

MATEUSZ DZIEDZIC, SZYMON ŁUKASIK*

METODA EWOLUCJI RÓŻNICOWEJ W KLASTERYZACJI Z OGRANICZENIAMI TWARDYMI

DIFFERENTIAL EVOLUTION METHOD IN HARD CONSTRAINED CLUSTERING

Streszczenie

Gwałtowny wzrost pojemności baz danych oraz znaczne zwiększenie wymiarowości rozpatrywanych zbiorów wymusza wykorzystanie do ich analizy skutecznych metod komputerowych. Głównym celem artykułu jest zaprezentowanie możliwości zastosowania oraz skuteczności ewolucji różnicowej w zadaniu klasteryzacji z ograniczeniami, które stanowi wariant jednego z najważniejszych zagadnień eksploracji danych. W artykule przedstawiono formalnie zadanie klasteryzacji z ograniczeniami oraz metodę ewolucji różnicowej wraz z jej implementacją. Zaprezentowano również wyniki badań eksperymentalnych, w tym dotyczących klasteryzacji zestawów przebiegów czasowych. Mogą być one zinterpretowane jako przykład grupowania pomiarów wartości elektrycznych.

Słowa kluczowe: ewolucja różnicowa, metody heurystyczne, eksploatacja danych, klasteryzacja z ograniczeniami

Abstract

Rapid increase of database capacity and high dimensionality of considered datasets forces the usage of effective computer methods to their analysis. The main goal of this paper is a presentation of possibility of employing and efficiency of Differential Evolution in the constrained clustering, a variant of one of the most important data mining tasks. In the paper the constrained clustering task and Differential Evolution method, with its implementation, was presented. Moreover, the results of the experimental studies were shown, including the clustering of time series set. It could be interpreted as an example of grouping of electrical variables measurement.

Keywords: Differential Evolution, heuristic methods, data mining, constrained clustering

* Mgr inż. Mateusz Dziedzic, mgr inż. Szymon Łukasik, Katedra Automatyki i Technik Informatycznych, Wydział Inżynierii Elektrycznej i Komputerowej, Politechnika Krakowska.

1. Wstęp

Eksploracja danych jest w ostatnich latach bardzo dynamicznie rozwijającą się dziedziną nauki. Najważniejszym jej celem jest znajdowanie niewidocznych dla człowieka (głównie z uwagi na ograniczone zasoby czasowe, możliwości obliczeniowe oraz wielowymiarowość rozpatrywanych zbiorów) prawidłowości oraz zależności o różnym charakterze. Jedną z ważniejszych technik analizy danych jest klasteryzacja będąca przedmiotem rozważań niniejszego artykułu.

Zadaniem procesu klasteryzacji jest podział zbioru obiektów na określoną liczbę klastrów z założeniem, by pojedynczy klaster zawierał obiekty do siebie w jak największym stopniu podobne, a równocześnie, by obiekty przypisane do różnych klastrów istotnie się między sobą różniły. Zastosowania klasteryzacji to przede wszystkim grupowanie obiektów np. dokumentów czy stron internetowych, a także wstępne przetwarzanie danych dla innych algorytmów umożliwiające zmniejszenie rozmiaru zbioru danych.

Uwzględnienie wiedzy dostępnej *a priori* w procesie klasteryzacji pozwala na znaczną poprawę jakości rozwiązania i jego dostosowanie do uwarunkowań problemu. Wynikiem tego jest sformułowanie nowego zagadnienia analizy danych, które w literaturze określane jest jako klasteryzacja z ograniczeniami. Za najczęściej stosowaną, klasyczną technikę klasteryzacji (również z ograniczeniami) uważa się algorytm *k*-średnich (ang. *k-means algorithm*), który jako istotny punkt odniesienia będzie w niniejszym artykule wielokrotnie przywoływany [10].

W dziedzinie automatyki i elektrotechniki klasteryzacja znajduje zastosowanie głównie w procedurach wykrywania uszkodzeń. Dane pochodzące z czujników mogą mieć charakter przebiegów czasowych wielkości elektrycznych pracujących napędów lub innych urządzeń, pomiarów temperatury itp. Dane takie są archiwizowane, również w czasie awarii lub innych nieprawidłowych stanów pracy maszyn. Pozwala to na przeprowadzenie procesu ich klasteryzacji, a następnie klasyfikacji przyszłych sygnałów i ich przypisanie do określonej grupy (na przykład praca normalna, uszkodzenie, awaria itp.).

Gwałtowny wzrost pojemności baz danych oraz znaczne zwiększenie wymiarowości rozpatrywanych zbiorów wymusza wykorzystanie do ich analizy metod komputerowych. Wraz z koniecznością zastosowania komputerów pojawia się potrzeba opracowania skutecznych metod oraz algorytmów o minimalnym zapotrzebowaniu na zasoby obliczeniowe, a także o możliwie jak największej efektywności i jakości otrzymywanych wyników. Znaczny rozmiar przestrzeni rozwiązań problemu klasteryzacji, nawet przy małej liczbie klastrów oraz jego istotna złożoność obliczeniowa powoduje, że w ciągu ostatnich 30 lat coraz częściej podejmowane są próby wykorzystania w tym zagadnieniu optymalizacyjnych metod heurystycznych (patrz np. [4, 5]). Nowoczesną i efektywną techniką należąca do tej klasy algorytmów, z powodzeniem stosowaną w omawianym zadaniu, jest ewolucja różnicowa [12, 17] (ang. *Differential Evolution*, w skrócie: DE).

Jednym z celów niniejszego artykułu jest przedstawienie zadania klasteryzacji z ograniczeniami, ze szczególnym uwzględnieniem jego zastosowania w zagadnieniach inżynierii elektrycznej. Następnie autor pragnie przedstawić opracowany i zaimplementowany w środowisku Matlab algorytm dedykowany dla tego problemu oparty o ewolucję róż-

nicową. Głównym przedmiotem rozważań będzie porównanie efektywności stworzonego algorytmu z klasyczną techniką k -średnich. Na podstawie przedstawionego porównania autor pragnie wskazać, że wykorzystanie ewolucji różnicowej w omawianym zagadnieniu można zdecydowanie uznać za celowe. Ambicją autora niniejszego artykułu jest także podanie cennych wskazówek ułatwiających użycie tej metody w zadaniu klasteryzacji z ograniczeniami oraz wskazanie możliwych kierunków przyszłych badań w tym zakresie.

2. Problem klasteryzacji z ograniczeniami twardymi

2.1. Formalizacja problemu klasteryzacji

Sformalizowanie problemu klasteryzacji stanowi zamieszczona poniżej definicja [6, 12]:

Definicja 1 (Klasteryzacja)

Niech zdefiniowany będzie zbiór m obiektów (punktów) n -wymiarowych \mathbf{X} oraz niech $\mathbf{X}_{m \times n}$ stanowi macierz o wymiarach $m \times n$ ów zbiór określająca. Każdy i -ty obiekt jest scharakteryzowany przez n -wymiarowy wektor \mathbf{x}_i ($i = 1, 2, \dots, m$), gdzie każdy element x_{ij} w \mathbf{x}_i odpowiada j -temu atrybutowi ($j = 1, 2, \dots, n$) i -tego obiektu.

Mając daną macierz $\mathbf{X}_{m \times n}$ celem algorytmu klasteryzującego jest określenie podziału

$$G = \{C_1, C_2, \dots, C_k\}, \quad (1)$$

(przy czym: $C_i \neq \emptyset$ dla każdego $i = 1, \dots, k$; $C_i \cap C_j = \emptyset$ dla każdego $i = 1, \dots, k, j = 1, \dots, k, i \neq j$ oraz $\bigcup_{i=1}^k C_i = \mathbf{X}$) z zachowaniem jak największej jednorodności wewnątrzklastrowej oraz jak największego zróżnicowania pomiędzy klastrami.

Proces klasteryzacji, jako zadania optymalizacji kombinatorycznej, polega na znalezieniu spośród zbioru wszystkich poprawnych podziałów $\mathbf{G} = \{G^1, G^2, \dots, G^l\}$ ($G^i \neq G^j, i \neq j$), takiego podziału G^* który minimalizuje pewną funkcję f określającą jego adekwatność:

$$\min_G f(\mathbf{X}_{m \times n}, \mathbf{G}) \quad (2)$$

2.2. Ograniczenia twarde

Podczas przeprowadzania procesu klasteryzacji często okazuje się, że jest dostępny pewien zakres wiedzy *a priori* dotyczący poprawnego (bądź pożądanego) podziału obiektów. Może to być wiedza ekspercka dotycząca konkretnego zagadnienia, jak również pewne preferencje osobiste, których spełnienie jest wymagane. Tak określone ograniczenia wprowadzane przez ową dodatkową wiedzę nazywamy ograniczeniami twardymi, w przeciwieństwie do ograniczeń miękkich, których spełnienie jest jedynie pożądane (z pewną określoną wagą) [11].

W niniejszym artykule rozważane są podstawowe rodzaje ograniczeń – ograniczenia łączności (ang. *must-link constraints*), zwane dalej ML oraz ograniczenia rozłączności (ang. *cannot-link constraints*), zwane dalej CL [21, 22].

Formalnie ograniczenia łączności oraz rozłączności definiowane są następująco [21, 22]:

Definicja 2 (Ograniczenia łączności i rozłączności)

Niech $\mathbf{Con}_= = \{Con_{=1}, Con_{=2}, \dots, Con_{=n_{ML}}\} \subseteq \mathbf{X} \times \mathbf{X}$ będzie zbiorem ograniczeń łączności (ML) oraz niech $\mathbf{Con}_\neq = \{Con_{\neq 1}, Con_{\neq 2}, \dots, Con_{\neq n_{CL}}\} \subseteq \mathbf{X} \times \mathbf{X}$ będzie zbiorem ograniczeń rozłączności (CL), przy czym $Con_{=i}$ oraz $Con_{\neq i}$ wiążą parę obiektów x i y , ($x \neq y$), dla których jest określone ograniczenie.

Liczba ograniczeń łączności dla zestawu danych określona jest jako n_{ML} :

$$n_{ML} = \sum_{i=1}^k \binom{n_i}{2} = \sum_{i=1}^k \frac{n_i!}{2!(n_i-2)!} = \frac{1}{2} \sum_{i=1}^k n_i(n_i-1), \quad (3)$$

gdzie:

- k – liczba wyróżnionych w zestawie klastrów,
- n_i – liczność i -tego klastra.

Natomiast liczba ograniczeń rozłączności n_{CL} jako:

$$n_{CL} = \sum_{i=1}^{k-1} n_i \sum_{j=i+1}^k n_j. \quad (4)$$

Łączną liczbę ograniczeń oznacza się jako:

$$n_{Con} = n_{ML} + n_{CL}. \quad (5)$$

Dla każdego $Con_{=i}$, jeżeli jeden z obiektów pary jest przypisany do klastra C_j , to również drugi z obiektów pary **musi** do niego należeć, co formalnie można zapisać następująco:

$$(x, y) \in Con_{=i} \wedge x \in C_j \Rightarrow y \in C_j \quad (6)$$

Dla każdego $Con_{\neq i}$, jeżeli jeden z obiektów pary jest przypisany do klastra C_j , to drugi z obiektów pary **nie może** do niego należeć. Formalnie:

$$(x, y) \in Con_{\neq i} \wedge x \in C_j \Rightarrow y \notin C_j \quad (7)$$

Aby w pełni wykorzystać dostępną wiedzę (oraz zapewnić jak najbardziej efektywne działanie badanych metod) na dostępnych zbiorach ograniczeń, dokonuje się *domknięcia przechodniego relacji*. Domknięcie przechodnie ograniczeń ML polega na utworzeniu nowego ograniczenia łączności $Con_{=i}$ dla każdej pary (x, y) , którą można połączyć przez dowolną kombinację już istniejących ograniczeń ML. Należy zwrócić uwagę, że – mimo iż jedynie zbiór ograniczeń ML *definiuje relację przechodnią*:

$$(x, y) \in \text{Con}_{=i} \wedge (y, z) \in \text{Con}_{=j} \Rightarrow (x, z) \in \text{Con}_{=h} \quad (8)$$

– to jednak procedurę *domknięcia przechodniego* wykonuje się również na zbiorze ograniczeń CL (z uwzględnieniem zbioru ML) z uwagi na zależność:

$$(x, y) \in \text{Con}_{=i} \wedge (y, z) \in \text{Con}_{\neq j} \Rightarrow (x, z) \in \text{Con}_{\neq h}. \quad (9)$$

3. Ewolucja różnicowa oraz metoda k -średnich

3.1. Metoda k -średnich

Jako pierwsza zostanie przedstawiona metoda klasyczna z modyfikacją uwzględniającą ograniczenia twarde. Została ona zaprezentowana po raz pierwszy przez MacQueen'a w artykule [10]. Jest to metoda wymagająca określenia ilości klastrów przed rozpoczęciem procesu. Algorytm 1 prezentuje procedurę k -średnich z zaimplementowanymi ograniczeniami twardymi [22]. Modyfikacją pozwalającą wprowadzić ograniczenia jest punkt 2 procedury **COPkmeans** – funkcja **violateC** – uwzględniająca warunek spełnienia ograniczeń podczas przypisywania punktów do klastrów.

Algorytm 1

Metoda k -średnich z zaimplementowanymi ograniczeniami twardymi

Procedura COP-kmeans

1. Wylosuj k początkowych centrów klastrów $\mu_1, \mu_2, \dots, \mu_k$
2. **Dla każdego** $x_i \in X$: przypisz x_i do najbliższego klastra C_j , dla którego zachodzi:
violateC = *false*
Zwróć \emptyset jeżeli taki klaster nie istnieje
3. **Dla każdego** $\mu_i : \mu_i = \left(\sum_{x \in C_i} x \right) / m_i$, gdzie m_i – ilość obiektów przypisanych do i -tego klastra
4. **Powtarzaj** (2) i (3) do osiągnięcia warunku stopu
5. **Zwróć** $\{\mu_1, \mu_2, \dots, \mu_k\}$

Funkcja violateC

1. **Dla każdego** $(x_i, x_j) \in \text{Con}_{=}$: Jeżeli $x_j \notin C_j$, **Zwróć** *true*
2. **Dla każdego** $(x_i, x_j) \in \text{Con}_{\neq}$: Jeżeli $x_j \in C_j$, **Zwróć** *true*
3. **W przeciwnym wypadku**: **Zwróć** *false*

Metoda k -średnich umożliwia znalezienie takiego podziału G^* , który minimalizuje (lokalnie) wskaźnik jakości [1, 2]:

$$f(\mathbf{X}_{m \times n}, G) = \sum_{i=1}^k \sum_{\mathbf{x} \in C_i} d(\mathbf{x}, \mu_i), \quad (10)$$

gdzie:

$$\mu_i = \left(\sum_{\mathbf{x} \in C_i} \mathbf{x} \right) / m_i \quad \begin{array}{l} \text{– wektor współrzędnych środka klastra } C_i, \text{ do którego przy-} \\ \text{pisano } m_i \text{ obiektów,} \end{array}$$

$$d(\mathbf{x}, \mu_i) \quad \begin{array}{l} \text{– przyjęta miara odległości (jeżeli nie zaznaczono inaczej,} \\ \text{w niniejszym artykule rozważana jest metryka euklidesowa).} \end{array}$$

Powyższa metoda stanowi w niniejszym artykule procedurę referencyjną pozwalającą ocenić efektywność oraz celowość zastosowania metody ewolucji różnicowej.

3.2. Ewolucja różnicowa

Ewolucja różnicowa jest optymalizacyjną metodą heurystyczną [17], która okazała się skuteczna w wielowymiarowych problemach optymalizacji funkcji o wartościach rzeczywistych [20]. Jest to iteracyjna metoda optymalizacyjna oparta na zaobserwowanych w przyrodzie mechanizmach ewolucyjnych – krzyżowaniu, mutacji oraz selekcji najlepszych osobników. Najważniejszą jej cechą, dającą przewagę nad klasycznymi algorytmami optymalizacji (np. gradientowymi), jest brak wymagań co do istnienia gradientu lub ciągłości funkcji optymalizowanych, co ma kluczowe znaczenie w przypadku zadania klasteryzacji.

Algorytm 2 przedstawia procedurę ewolucji różnicowej w uproszczonej wersji bez uwzględnienia rodzaju operatorów mutacji oraz krzyżowania [12].

Algorytm 2

Ewolucja różnicowa

Procedura COP-DE

1. **Przyjmij numer iteracji** $t \leftarrow 0$
 2. **Zainicjuj populację** pop_0 Np losowymi osobnikami x_1, x_2, \dots, x_{Np}
 3. **Dla każdego** $x_i \in pop_0$ oblicz wartość kryterium oceny $f(x_i)$
 4. **Powtarzaj do osiągnięcia warunku stopu:**
 - 4.1. $t++$
 - 4.2. **Dla każdego** $x_i \in pop_{t-1}$
 - 4.2.1. Mutacja \rightarrow stwórz wektor różnicowy y_i
 - 4.2.2. Krzyżowanie różnicowe pomiędzy x_i i $y_i \rightarrow$ utwórz kandydata c_i
 - 4.2.3. Oblicz wartość kryterium oceny $f(c_i)$
 - 4.2.4. Selekcja: **Jeżeli** $f(c_i) > f(x_i) \rightarrow pop_t(i) = c_i$
w przeciwnym wypadku $pop_t(i) = x_i$
 5. **Zwróć** najlepszego osobnika x_{best}
-

Ewolucja różnicowa jest metodą prostą w implementacji głównie z uwagi na małą liczbę parametrów. Pomijając standardowe parametry tej klasy algorytmów – wielkość

populacji Np oraz maksymalną liczbę generacji t_{\max} (określaną nawet w przypadku stosowania innego kryterium stopu) – ewolucja różnicowa operuje dwoma parametrami (współczynnikami): mutacji $F \in \mathbf{R}$, charakteryzującym wielkość wpływu wektora różnicowego na rodzica głównego podczas generowania wektora różnicowego y_i , oraz krzyżowania $Cr \in \mathbf{R}$, określającym prawdopodobieństwo jego wystąpienia dla każdego atrybutu.

W rozpatrywanym zadaniu klasteryzacji z ograniczeniami osobnik (w tym rozwiązanie, które stanowi najlepszy ze znalezionych osobników) reprezentowany jest, podobnie jak w przypadku metody k -średnich, poprzez zbiór k centrów klastrów $\{\mu_1, \mu_2, \dots, \mu_k\}$. Opcjonalnie możliwe jest wygenerowanie przyporządkowania wszystkich obiektów najbliższemu centrum klastrów z zachowaniem warunku spełnienia wszystkich ograniczeń.

3.2.1. Warianty ewolucji różnicowej

Wariantami ewolucji różnicowej określa się zwykle różne implementacje metody z konkretnymi operatorami mutacji i krzyżowania. Najogólniejszy podział operatorów mutacji zakłada wyszczególnienie dwóch grup: operatorów deterministycznych, w których wartość parametru F jest stała, oraz adaptacyjnych, wykorzystujących losowy dobór parametru z przedziału zależnego m.in. od numeru generacji lub aktualnej wartości (czy też wielkości zmian) funkcji oceny.

Spośród licznych możliwych wariantów ewolucji różnicowej w niniejszym artykule zaprezentowano wyniki uzyskane dla następujących wariantów:

– deterministycznych:

– **DE/rand/1** [17]:

Wektor różnicowy y_i tworzony jest następująco:

$$y_i = x_{r_1} + F(x_{r_2} - x_{r_3}), \quad (11)$$

gdzie $r_1, r_2, r_3 \in \{1, 2, \dots, Np\} \wedge (r_1 \neq r_2 \neq r_3 \neq i)$ oraz $F \in [0, 2]$.

– **DERLs** [8]:

$$y_i = x_{t_b} + F(x_{t_2} - x_{t_3}), \quad (12)$$

gdzie x_{t_b} jest osobnikiem najlepszym (w sensie wartości kryterium oceny) spośród trzech losowych rodziców $x_{r_1}, x_{r_2}, x_{r_3}$, natomiast x_{t_2}, x_{t_3} są pozostałymi osobnikami z wylosowanej trójki i ponadto: $r_1, r_2, r_3 \in \{1, 2, \dots, Np\} \wedge (r_1 \neq r_2 \neq r_3 \neq i)$ oraz $F \in [0, 2]$.

– adaptacyjnych:

– **Dither, Jitter** [13, 14]:

Idea obu wariantów sprowadza się do losowego doboru parametru F dla każdego z osobników populacji.

W pierwszym z wymienionych wariantów parametr F otrzymywany jest według następującego rozkładu:

$$F \sim U(F_{\min}, F_{\max}), \quad (13)$$

W wariancie Jitter natomiast parametr F jest traktowany jako wektor wartości losowych F_p , $i = 1, \dots, n$ losowany według rozkładu:

$$F_i \sim U_i(F_{\min}, F_{\max}). \quad (14)$$

W obu wariantach wektor różnicowy y_i tworzony jest na podstawie wybranej procedury stosowanej w wariantach deterministycznych (tu DE/rand/1).

W każdym przypadku zastosowano wykładniczy operator krzyżowania (ang. *exponential crossover*) [17]. Kandydat c_p , który ma zostać porównany z osobnikiem x_i tworzony jest według następującego schematu:

$$c_{ij} = \begin{cases} y_{ij} & \text{dla } j = \{\langle p \rangle_n, \langle p+1 \rangle_n, \dots, \langle p+L-1 \rangle_n\} \\ x_{ij} & \text{w przeciwnym wypadku} \end{cases}. \quad (15)$$

gdzie:

$\langle \cdot \rangle_n$ – oznacza działanie modulo n , czyli resztę z dzielenia przez n .

Pozycja początkowa p wybierana jest losowo ze zbioru $\{1, 2, \dots, n\}$, parametr L reprezentujący długość wymienianego w procesie krzyżowania fragmentu wektora różnicowego przyjmuje wartość z tego samego przedziału zbioru losowaną według wykładniczego rozkładu prawdopodobieństwa: $P(L) = (Cr)L^{-1}$. Przykład ilustrujący krzyżowanie z zastosowaniem tego operatora można znaleźć między innymi w pracy [17].

Wybór operatora krzyżowania został podyktowany wynikami badań wstępnych, w których uzyskano nieznacznie lepsze wyniki przy zastosowaniu tego operatora. Badania te potwierdziły również opinię Storna o znikomym wpływie rodzaju operatora krzyżowania na wyniki [20].

Przedmiotem rozważań w toku badań eksperymentalnych były różne modyfikacje ewolucji różnicowej. Między innymi przebadano szereg wariantów deterministycznych zaproponowanych przez Storna [18]. Znaczną uwagę poświęcono również wariantom adaptacyjnym: dwóm przedstawionym wcześniej – **Dither** oraz **Jitter**, jak również zaproponowanemu przez Dasa **DETVSF** [3] w dwóch wersjach (przedział losowania parametru F uzależniono od numeru iteracji w pierwszej oraz najwyższej i najniższej wartości funkcji oceny osobników aktualnego pokolenia w drugiej z nich) czy też **DERL** [8]. Do prezentacji w niniejszym artykule wybrano warianty najprostsze w implementacji oraz najmniej skomplikowane obliczeniowo, dające mimo wszystko bardzo dobre rezultaty.

3.2.3. Funkcja oceny jakości osobnika

Selekcja osobników stanowi sedno wszystkich metod ewolucyjnych. Celem tych metod jest poprawianie jakości kolejnych populacji poprzez wybór najlepszych osobników populacji obecnej (rodzicielskiej) i przekazanie ich do populacji kolejnej (potomnej). Proces ten w ewolucji różnicowej polega na porównaniu wszystkich par osobników (obecnego osobnika x_i oraz kandydata c_i) i wybraniu lepszego z nich do populacji potomnej. Konieczne zatem jest zdefiniowanie funkcji umożliwiającej taką ocenę. W niniejszym artykule jako funkcję oceny jakości osobników przyjęto kryterium TRW [12]. Zdefiniowanie wspomnianego kryterium wymaga na wstępie wprowadzenia pojęcia *macierzy rozrzutu wewnątrzklastrowego* \mathbf{W} :

$$\mathbf{W}(\mathbf{X}_{m \times n}, G) = \sum_{i=1}^k \sum_{\mathbf{x} \in C_i} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_i)' (\mathbf{x}^{(i)} - \boldsymbol{\mu}_i), \quad (16)$$

gdzie $\boldsymbol{\mu}_i$ jak w równaniu (10).

Kryterium TRW zdefiniowane jest jako:

$$\min_G \text{tr}(\mathbf{W}), \quad (17)$$

Wskaźnik ten pozwala na bezpośrednie porównanie efektywności ewolucji różnicowej z metodą k -średnich, gdyż poprzez trywialne przekształcenia matematyczne można wykazać, iż jego wartość jest proporcjonalna do wartości wskaźnika jakości zdefiniowanego według wzoru (10).

3.2.4. Warunek zatrzymania pracy algorytmu

Obiektywne porównanie efektywności badanych metod zapewnia udostępnienie im jednakowej ilości zasobów. Za tzw. wąskie gardło algorytmu w większości opracowań (m.in. [12]) uważa się określenie jakości osobnika (obliczenie funkcji oceny). Zgodnie z tą obserwacją w badaniach jako warunek stopu ewolucji różnicowej została zastosowana maksymalna ilość iteracji, która wraz z odpowiednio dobraną liczebnością populacji wymagała wykonania 210 obliczeń funkcji oceny osobników. Ilość osobników $Np = 10$ oraz ilość iteracji $t_{\max} = 20$ dobrano doświadczalnie, powyżej tych wartości wzrost jakości wyników był znikomy, natomiast znacznemu wydłużeniu ulegał czas obliczeń. Dla powyższych parametrów funkcja oceny osobników jest obliczana dziesięciokrotnie podczas fazy inicjalizacji oraz po 10 razy w każdej iteracji. W przypadku metody k -średnich ustawiono limit iteracji na 210 odpowiadający 210 obliczeniom funkcji oceny osobników (w każdej iteracji metody k -średnich jednokrotnie przeprowadzane są operacje odpowiadające pojedynczej ewaluacji funkcji oceny osobnika w ewolucji różnicowej).

4. Wyniki badań eksperymentalnych

4.1. Prezentacja zbiorów danych

Jak wspomniano wcześniej zadanie klasteryzacji z ograniczeniami twardymi polega na podziale zbioru obiektów (danych) na określoną ilość klastrów z zachowaniem wszystkich ograniczeń oraz maksymalizacją wskaźnika jakości klasteryzacji. W niniejszej pracy do zbadania efektywności stworzonego algorytmu wykorzystano 6 typów pseudolosowo generowanych zestawów danych oraz rzeczywiste zestawy pozyskane z repozytorium *UCI Machine Learning Repository* [7].

Przed przystąpieniem do klasteryzacji zestawy danych zostały wstępnie przetworzone: zmapowano metodą „1 z n ” [15] nieliczbowe atrybuty zestawów rzeczywistych oraz znormalizowano do przedziału $[0, 1]$ wartości atrybutów wszystkich wykorzystanych zestawów danych.

4.1.1. Zestawy wygenerowane pseudolosowo

Na potrzeby niniejszego artykułu wygenerowano 5 zestawów z każdego z 6 typów następujących trójwymiarowych zbiorów danych pseudolosowych:

- **a1** – 3 kuliste klastry o średnim stopniu łączności,
- **a2** – 3 kuliste klastry o dużym stopniu łączności,
- **a3** – 5 kulistych klastrów o średnim stopniu łączności,
- **a4** – 5 kulistych klastrów o dużym stopniu łączności,
- **a5** – 3 elipsoidalne klastry o średnim stopniu łączności,
- **a6** – 3 elipsoidalne klastry o dużym stopniu łączności.

Tabela 1

Charakterystyka zestawów danych pseudolosowych

Zestaw	Macierz	Rozkład	Parametry
a1	$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \mathbf{X}_3 \end{bmatrix}$	$\mathbf{X}_1 \sim N(\mu_1, \sigma_1) (30 \times 3)$ $\mathbf{X}_2 \sim N(\mu_2, \sigma_2) (30 \times 3)$ $\mathbf{X}_3 \sim N(\mu_3, \sigma_3) (30 \times 3)$	$\mu_1 = [0, 0, 0], \sigma_1 = 1, 0$ $\mu_2 = [2, 2, 2], \sigma_2 = 1, 0$ $\mu_3 = [4, 4, 4], \sigma_3 = 1, 0$
a2	$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \mathbf{X}_3 \end{bmatrix}$	$\mathbf{X}_1 \sim N(\mu_1, \sigma_1) (30 \times 3)$ $\mathbf{X}_2 \sim N(\mu_2, \sigma_2) (30 \times 3)$ $\mathbf{X}_3 \sim N(\mu_3, \sigma_3) (30 \times 3)$	$\mu_1 = [0, 0, 0], \sigma_1 = 1, 5$ $\mu_2 = [2, 2, 2], \sigma_2 = 1, 5$ $\mu_3 = [4, 4, 4], \sigma_3 = 1, 5$
a3	$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_5 \end{bmatrix}$	$\mathbf{X}_1 \sim N(\mu_1, \sigma_1) (30 \times 3)$ \vdots $\mathbf{X}_5 \sim N(\mu_5, \sigma_5) (30 \times 3)$	$\mu_1 = [0, 0, 0], \sigma_1 = 1, 0$ $\mu_2 = [2, 2, 2], \sigma_2 = 1, 0$ $\mu_3 = [4, 4, 4], \sigma_3 = 1, 0$ $\mu_4 = [4, 0, 4], \sigma_4 = 1, 0$ $\mu_5 = [0, 4, 0], \sigma_5 = 1, 0$
a4	$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_5 \end{bmatrix}$	$\mathbf{X}_1 \sim N(\mu_1, \sigma_1) (30 \times 3)$ \vdots $\mathbf{X}_5 \sim N(\mu_5, \sigma_5) (30 \times 3)$	$\mu_1 = [0, 0, 0], \sigma_1 = 1, 5$ $\mu_2 = [2, 2, 2], \sigma_2 = 1, 5$ $\mu_3 = [4, 4, 4], \sigma_3 = 1, 5$ $\mu_4 = [4, 0, 4], \sigma_4 = 1, 5$ $\mu_5 = [0, 4, 0], \sigma_5 = 1, 5$
a5	$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \mathbf{X}_3 \end{bmatrix}$	$\mathbf{X}_1 \sim N(\mu_1, \sigma_1) (30 \times 3)$ $\mathbf{X}_2 \sim N(\mu_2, \sigma_2) (30 \times 3)$ $\mathbf{X}_3 \sim N(\mu_3, \sigma_3) (30 \times 3)$	$\mu_1 = [0, 0, 0], \sigma_1 = 6, 0$ $\mu_2 = [0, 2, 0], \sigma_2 = 1, 0$ $\mu_3 = [0, 4, 2], \sigma_3 = 1, 0$
a6	$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \mathbf{X}_3 \end{bmatrix}$	$\mathbf{X}_1 \sim N(\mu_1, \sigma_1) (30 \times 3)$ $\mathbf{X}_2 \sim N(\mu_2, \sigma_2) (30 \times 3)$ $\mathbf{X}_3 \sim N(\mu_3, \sigma_3) (30 \times 3)$	$\mu_1 = [0, 0, 0], \sigma_1 = 9, 0$ $\mu_2 = [0, 2, 0], \sigma_2 = 1, 5$ $\mu_3 = [0, 4, 2], \sigma_3 = 1, 5$

Do wygenerowania poszczególnych zestawów wykorzystano wielowymiarowy generator liczb pseudolosowych o rozkładzie normalnym, którego szczegółowe parametry dla poszczególnych przypadków przedstawiono w tabeli 1.

4.1.2. Zestawy wygenerowane pseudolosowo

Aby zbadać efektywność wybranych metod w stosunku do rzeczywistych problemów klasteryzacji wykorzystano 5 zestawów danych dostępnych w *UCI Machine Learning Repository* [7]. Opis oraz charakterystykę zestawów prezentuje tabela 2.

Tabela 2

Charakterystyka i źródła zestawów danych rzeczywistych				
Zestaw	k	n	m	Nazwa i źródło
r1	3	4	150	Iris Data Set http://archive.ics.uci.edu/ml/datasets/Iris
r2	2	9	683	Breast Cancer Wisconsin (Diagnostic) DS http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)
r3	2	9	958	Tic-Tac-Toe Endgame Data Set http://archive.ics.uci.edu/ml/datasets/Tic-Tac-Toe+Endgame
r4	6	4	4435	Statlog (Landsat Satellite) Data Set http://archive.ics.uci.edu/ml/datasets/Statlog+(Landsat+Satellite)
r5	6	30	600	Synthetic Control Chart Time Series http://archive.ics.uci.edu/ml/datasets/Synthetic+Control+Chart+Time+Series

k – ilość klastrów,

n – ilość atrybutów (wymiar) zestawu danych,

m – liczność oryginalnego zestawu danych.

W celu ograniczenia liczności klasteryzowanych zbiorów (z uwagi na dostępne zasoby sprzętowe) z zestawów **r2** – **r5** pięciokrotnie (w celu zminimalizowania wpływu losowości na wyniki) wybrano w sposób pseudolosowy zbiór 100 obiektów, które stanowiły zbiory klasteryzowane.

Spośród zastosowanych zestawów rzeczywistych na szczególną uwagę zasługuje zestaw **r5** – Synthetic Control Chart Time Series, stanowiący zbiór 600 sztucznie wygenerowanych przebiegów czasowych (6 klastrów po 100 przebiegów), które mogą być zinterpretowane jako wyniki pomiarów wartości elektrycznych (na przykład pomiary wartości napięć i prądów reprezentujące różne stany pracy urządzeń elektrycznych, włączając uszkodzenia i awarie, których jak najwcześniejsze wykrywanie jest kluczowym zadaniem w inżynierii). Do jego klasteryzacji wykorzystana została miara odległości oparta o nieliniową transformację czasową (ang. *Dynamic Time Warping* – DTW) [9].

4.1.3. Generowanie ograniczeń

W ramach niniejszego artykułu rozpatrywane jest zadanie klasteryzacji z ograniczeniami, konieczne zatem jest wygenerowanie ich na podstawie dostępnych podziałów wzorcowych. Każde badanie poprzedzone jest wygenerowaniem zbiorów ograniczeń łączności $Con_{\underline{}}$ i rozłączności $Con_{\overline{}}$. Każdorazowo losowany jest określony procent wszystkich możliwych ograniczeń.

W procesie generowania ograniczeń wybierana jest losowo para obiektów, następnie na podstawie ich przynależności w podziale wzorcowym para dodawana jest do określonego zbioru ograniczeń aż do otrzymania odpowiedniej ich liczby.

4.2. Ocena zgodności klasteryzacji – indeks Randa

W celu miarodajnego zestawienia efektywności używanych w pracy metod klasteryzacji wprowadzone zostanie pojęcie indeksu Randa [16] pozwalające na określenie podobieństwa dwóch rozwiązań zadania klasteryzacji.

Mając dany m -elementowy zbiór obiektów $\mathbf{X} = \{x_1, x_2, \dots, x_m\}$ i dwa porównywane podziały tego zbioru, G^1 oraz G^2 , definiuje się następujące wartości:

- a – liczba par obiektów zbioru \mathbf{X} , które znajdują się w jednym klastrze w G^1 oraz G^2 (tak zwana decyzja prawdziwie pozytywna);
- b – liczba par obiektów zbioru \mathbf{X} , które znajdują się w różnych klastrach w G^1 oraz G^2 (tak zwana decyzja prawdziwie negatywna);
- c – liczba par obiektów zbioru \mathbf{X} , które znajdują się w jednym klastrze w G^1 oraz w różnych klastrach w G^2 (tak zwana decyzja fałszywie negatywna);
- d – liczba par obiektów zbioru \mathbf{X} , które znajdują się w różnych klastrach w G^1 oraz w jednym klastrze w G^2 (tak zwana decyzja fałszywie pozytywna).

Można to formalnie zapisać jako:

$$a = |S|, \text{ gdzie } S = \{(x, y) : x \neq y; x, y \in C_i^1; x, y \in C_j^2\} \quad (18)$$

$$b = |S|, \text{ gdzie } S = \{(x, y) : x \neq y; x \in C_i^1; y \in C_j^1; x \in C_k^2; y \in C_l^2\} \quad (19)$$

$$c = |S|, \text{ gdzie } S = \{(x, y) : x \neq y; x, y \in C_i^1; x \in C_i^2; y \in C_j^2\} \quad (20)$$

$$d = |S|, \text{ gdzie } S = \{(x, y) : x \neq y; x \in C_i^1, y \in C_j^1; x, y \in C_j^2\} \quad (21)$$

Na podstawie powyższych oznaczeń Indeks Randa R definiowany jest następująco:

$$R = \frac{a+b}{a+b+c+d} = \frac{a+b}{\binom{m}{2}} = 2 \cdot \frac{a+b}{m \cdot (m-1)}. \quad (22)$$

Indeks ten przyjmuje wartości w przedziale $[0,1]$. Bardziej intuicyjne wydaje się jego przeliczenie na zakres procentowy:

$$R = 2 \cdot \frac{a+b}{m \cdot (m-1)} \cdot 100\%. \quad (23)$$

Dzięki temu zabiegowi uzyskane wyniki przyjmują wartości 0–100%, co naturalnie można zinterpretować jako procent zgodności pomiędzy podziałami G^1 oraz G^2 .

W niniejszym artykule podział G^2 oznaczać będzie wzorcowy podział obiektów otrzymany w czasie generacji zestawów pseudolosowych lub pobrany razem z zestawami rzeczywistymi.

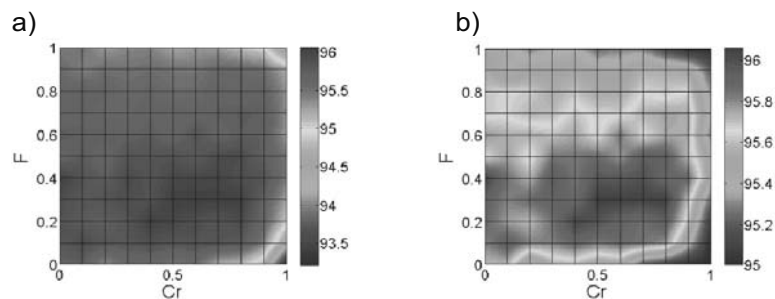
4.3. Dobór parametrów F i Cr

Dla celów niniejszego artykułu zbadano wpływ wartości parametrów DE na otrzymywane wyniki (dla $Cr = [0,0; 0,1; \dots; 1,0]$ oraz $F = [0,0; 0,1; \dots; 1,0]$). Na podstawie wyników otrzymanych dla wariantów **DE/rand/1/exp** oraz **DERLs/exp** dla zestawów pseudolosowych **a1** – **a6**, patrz: rys. 1, wyznaczono sugerowane wartości parametrów wykorzystane w dalszych badaniach:

- $F = 0,3$;
- $Cr = 0,8$.

Wartości te, w oparciu o propozycje wartości zaprezentowane przez Storna w pracy [19], wskazują na brak przedwczesnej zbieżności ewolucji różnicowej w rozważanym zadaniu klasteryzacji z ograniczeniami zaprezentowanych zestawów danych.

Jak wykazały badania dobór wartości parametrów ma marginalne znaczenie na jakość wyników, praktycznie dla całego przebadanego zakresu wyniki przekraczały 95%, osiągając najlepszy wynik 96,05%.



Rys. 1. a) Uśrednione wartości indeksu Randa dla 6 zestawów danych pseudolosowych; b) Wykres z obcięzonymi wartościami poniżej 95% – większa czytelność okolic maksimum

Fig. 1. a) Averaged Rand index values for 6 artificial data sets; b) Figure with values limited to >95% better readability of the maximum proximity

Analiza przedstawionych wykresów pozwala stwierdzić, iż parametr F ma większy wpływ na efektywność ewolucji różnicowej niż parametr Cr (taką samą zależność określili Storn i Price m.in. w [17, 19]). Z tego powodu w przypadku wariantów adaptacyjnych skoncentrowano się jedynie na sposobie wyboru pierwszego z nich uznając, iż niewłaściwie dobrana wartość Cr nie będzie miała znaczącego wplywu na wynik klasteryzacji.

4.4. Analiza porównawcza wyników klasteryzacji

Wyniki klasteryzacji dla wybranych wariantów DE oraz metody k -średnich dla wygenerowanych zestawów danych losowych i wybranych zestawów danych rzeczywistych przedstawiono w tabeli 3. Zestawiono w niej wartości średnie indeksu Randa wraz z jego średnimi odchyleniami standardowymi. Dodatkowo dla metody k -średnich przedstawiono procentową ilość powtórzeń zakończonych niepowodzeniem (dla DE nie zaobserwowano takich powtórzeń). Tabela 4 prezentuje procentową różnicę wartości średniej oraz średniego odchylenia standardowego indeksu Randa pomiędzy dwoma porównywanymi metodami, przy czym wartości otrzymane za pomocą metody k -średnich stanowią poziom odniesienia (0%), a $\Delta R_{\mu, \%}$ oznacza procentową zmianę średniej wartości indeksu Randa uśrednioną dla wszystkich zbadanych wariantów DE, czyli:

$$\Delta R_{\mu, \%} = \left(\frac{R_{\mu, DE}}{R_{\mu, k-means}} - 1 \right) \cdot 100. \quad (24)$$

Analogicznie $DR_{s, \%}$ określa procentową zmianę średniej wartości odchyżeń standardowych indeksu Randa uśrednioną jak wyżej:

$$\Delta R_{\sigma, \%} = \left(\frac{R_{\sigma, DE}}{R_{\sigma, k-means}} - 1 \right) \cdot 100\%. \quad (25)$$

Tabela 3

Wyniki klasteryzacji dla wybranych wariantów ewolucji różnicowej oraz metody k -średnich

Zestaw	DE/rand/l/exp		DERLs/exp		Dither/exp		Jitter/exp		k -średnich		
	R_{μ}	R_{σ}	R_{μ}	R_{σ}	R_{μ}	R_{σ}	R_{μ}	R_{σ}	R_{μ}	R_{σ}	<i>err</i>
a1	97,84	1,39	97,50	1,63	97,82	1,61	96,38	1,89	79,37	31,27	18,93
a2	94,61	1,65	94,38	1,72	94,30	1,73	92,89	2,17	66,86	24,73	28,80
a3	99,56	0,52	99,47	0,56	99,53	0,57	98,90	0,92	90,51	14,46	9,03
a4	95,82	0,94	95,99	0,92	96,16	0,78	95,55	0,95	80,42	17,23	16,27
a5	95,59	1,92	95,30	2,20	95,70	2,19	93,74	2,95	67,96	29,71	28,99
a6	92,46	2,10	91,86	2,85	91,62	2,41	91,08	12,92	59,44	23,90	35,77
μ_a	95,98	1,42	95,75	1,65	95,86	1,55	94,76	3,63	74,09	23,55	22,97
r1	98,37	0,72	98,28	0,79	98,47	0,62	97,93	0,98	66,69	12,55	28,76
r2	98,38	1,29	99,14	0,54	98,40	0,87	97,58	1,70	85,30	18,50	13,10
r3	87,20	3,24	86,87	3,85	87,39	2,62	87,39	2,93	60,56	17,25	31,47
r4	93,69	1,83	93,59	2,04	93,53	1,87	92,91	1,97	92,72	4,04	0,30
r5	88,59	2,68	88,15	2,73	93,67	1,74	95,78	1,36	85,27	1,87	0,30
μ_r	93,25	1,95	93,21	1,99	94,29	1,54	94,32	1,79	78,36	10,88	14,79
μ_{err}	94,79	1,69	94,48	1,82	95,08	1,55	94,54	2,71	76,23	17,22	18,88

Dla algorytmu DE uśredniony wzrost wartości średniej indeksu Randa dla wszystkich przypadków danych losowych wynosi $\sim 31,12\%$, natomiast odchylenie standardowe zmniejsza się średnio o $\sim 92,25\%$, co jest wynikiem bardzo dobrym. Nieco mniejsza poprawa wyników dla zbiorów rzeczywistych występuje głównie dla zestawów **r4** i **r5**, które okazały się łatwe do klasteryzacji metodą k -średnich. Niemniej zauważalna jest wciąż poprawa jakości wyników ich klasteryzacji po zastosowaniu metody ewolucji różnicowej. Dla zestawu **r4** jest to zmniejszenie wartości odchyżeń standardowych otrzy-

wanych wartości indeksu Randa o około 51,5% przy pomijalnej zmianie ich wartości średnich, natomiast w przypadku zestawu **r5** wartości otrzymanych wyników uległy poprawie o niecałe 8,5% przy porównywalnym wzroście odchyleń standardowych, co przy założeniu wielokrotnego uruchomienia algorytmu pozwala na otrzymanie lepszego rozwiązania.

Tabela 4

Zestawienie procentowe wyników klasteryzacji metodami ewolucji różnicowej oraz k -średnich

Zestaw	$DR_{m,\%}$	$DR_{s,\%}$
a1	22,78	-94,76
a2	40,71	-92,48
a3	9,87	-95,83
a4	19,29	-94,75
a5	39,98	-92,42
a6	54,09	-83,24
μ_a	31,12	-92,25
r1	49,71	-94,29
r2	15,21	-93,12
r3	39,13	-83,12
r4	0,73	-51,56
r5	8,37	7,18
μ_r	22,63	-62,98
$\mu_{a,r}$	26,88	-77,62

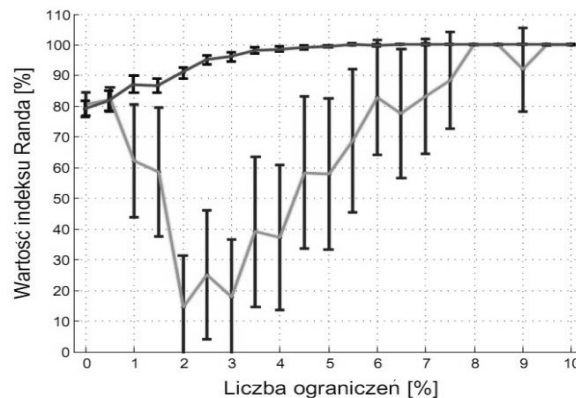
Zastanawiające mogą być osiągi wariantów adaptacyjnych DE, które praktycznie nie odbiegają od wyników uzyskiwanych dla wariantów deterministycznych. Jest to spowodowane najprawdopodobniej charakterystyką metody, która iteracyjnie zawęża przeszukiwaną aktualnie przestrzeń. To powoduje, że stosowanie wariantów modyfikujących przedział doboru parametru F w zależności od iteracji bądź aktualnych wartości funkcji oceny (na przykład warianty **DETVSF** i **DEFVSF** [3]) staje się niepotrzebne. Proces ten jest związany z długością wektora różnicowego, którego moduł dla kolejnych iteracji staje się coraz krótszy w sensie przyjętej metryki (poszczególne osobniki populacji znajdują się bliżej rozwiązania optymalnego, tak więc wektor powstały z ich różnicy ma mniejszą długość). Metoda ta w naturalny sposób implementuje techniki zwiększające zasięg eksploracji na początku procesu, aby na jego końcu przystąpić do szczegółowego przeszukiwania lokalnego.

Bardzo dobre wyniki klasteryzacji dla zestawu **r5** (wygenerowanych przebiegów czasowych) dla ewolucji różnicowej (prawie 96% zgodności przy wariancie **Jitter**), wskazują na możliwość zastosowania metody na przykład do klasteryzacji rzeczywistych przebiegów czasowych danych pomiarowych. Może to być wstępem do określenia różnych typów

przebiegów, identyfikacji odpowiadających im zdarzeń/stanów urządzenia oraz późniejszej klasyfikacji nowych danych pomiarowych.

4.5. Analiza wpływu liczby ograniczeń

Na rysunku 2 zaprezentowano uśrednione wyniki klasteryzacji badanych zestawów danych (**a1** – **a6** oraz **r1** – **r5**) przedstawione w formie wykresów wartości średniej i odchylenia standardowego w funkcji ilości ograniczeń.



Rys. 2. Wartości średnie oraz odchylenia standardowe wartości indeksu Randa uśrednione dla wszystkich zestawów danych

Fig. 2. Means and standard deviations of the Rand index averaged for all data sets

Analizując zamieszczone wyniki, stwierdzono, że metoda k -średnich sprawdza się porównywalnie z metodą DE jedynie dla stosunkowo dużej liczby ograniczeń (powyżej 10% n_{con} , co przykładowo stanowi 670 znanych ograniczeń dla zbioru **r1**). Ewolucja różnicowa stanowczo wykazuje swą wyższość, szczególnie pod względem stabilności (brak przypadków nieotrzymania poprawnego rozwiązania). Warte podkreślenia jest wysoka efektywność wariantu **Dither**, który jest najprostszym z zastosowanych wariantów adaptacyjnych.

5. Wnioski

W niniejszym artykule przedstawiono koncepcję zastosowania metody ewolucji różnicowej w procesie klasteryzacji z ograniczeniami. Zaprezentowano przykłady problemów opartych o wygenerowane zestawy pseudolosowe oraz rzeczywiste zbiory danych, porównano ponadto efektywność i jakość otrzymywanych wyników metody DE z wynikami uzyskanymi dla klasycznego algorytmu k -średnich. Otrzymane wyniki pozwalają uznać za zasadne wykorzystanie nowoczesnych metod heurystycznych (w szczególności metody ewolucji różnicowej) w rozpatrywanym problemie.

Wyniki uzyskane dla zestawu danych w postaci przebiegów czasowych zachęcają do dalszych badań w tym kierunku, szczególnie pod kątem minimalizacji wymaganych

zasobów, zwłaszcza czasowych. Dalsze badania związane będą z próbami uproszczenia procedury ewolucji różnicowej, co będzie miało znaczący wpływ na efektywność obliczeniową metody. Wyniki otrzymane dla najprostszych wariantów DE pozwalają sądzić, że nie powinno mieć to znaczącego wpływu na jakość uzyskiwanych wyników. Zastosowaną w badaniu metodę ewolucyjną można również przy stosunkowo małym nakładzie pracy zrównoleglić, co niewątpliwie stanowi interesujący kierunek dalszych prac.

Literatura

- [1] Basu S., Davidson I., Wagstaff K., *Constrained Clustering: Advances in Algorithms, Theory and Applications*, Chapman and Hall/CRC, 2009.
- [2] Bilenko M., Basu S., Mooney J., *Integrating Constraints and Metric Learning in Semi-Supervised Clustering*, Proceedings of the Twenty-First International Conference on Machine Learning, 2004, 11-18.
- [3] Das S., Abraham A., Konar A., *Particle Swarm Optimization and Differential Evolution Algorithms: Technical Analysis, Applications and Hybridization Perspectives*, [w:] *Advances of Computational Intelligence in Industrial Systems*, pod red. Liu Y., Sun A., Loh H.T., Lu W.F., Lim E.-P., Springer, 2008, 1-38.
- [4] Demiriz A., Bennett K.P., Embrechts M.J., *Semi-Supervised Clustering Using Genetic Algorithms*, Proceedings of Artificial Neural Networks in Engineering (ANNIE'99), 1999, 809-814.
- [5] Dyer M., Frieze A.M., *A Simple Heuristic for the p-Centre Problem*, Operations Research Letter 3(6), 1985, 285-288.
- [6] Ferligoj A., Batagelj V., *Clustering with Relational Constraint*, Psychometrika 47(4), 1982, 413-426.
- [7] Frank A., Asuncion A., *UCI Machine Learning Repository*, [online] 2010-11-30, (dostępny w Internecie: <http://archive.ics.uci.edu/ml/>).
- [8] Kaelo P., Ali M.M., *A numerical study of some modified differential evolution algorithms*, European Journal of Operational Research, 169(3), 2006, 1176-1184.
- [9] Keogh E.J., Pazzani M.J., *Derivative Dynamic Time Warping*, Proceedings of First SIAM International Conference on Data Mining, 2001, 187-194.
- [10] MacQueen J.B., *Some methods for classification and analysis of multivariate observations*, Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, 1967, 281-297.
- [11] Michalewicz Z., Fogel D.B., *How to Solve it: Modern Heuristics*, Springer, 2000.
- [12] Paterlini S., Krink T., *Differential evolution and particle swarm optimization in partitional clustering*, Computational Statistics & Data Analysis 50(5), 2006, 1120-1247.
- [13] Pedersen M.E.H., *Tuning & Simplifying Heuristical Optimization*, praca doktorska, University of Southampton, 2010.
- [14] Price K., Storn R.M., Lampinen J.A., *Differential Evolution: A Practical Approach to Global Optimization*, Springer, 2005.

- [15] Pyle D., *Data Preparation for Data Mining*, Morgan Kaufman, 1999.
- [16] Rand W.M., *Objective Criteria for the Evaluation of Clustering Methods*, Journal of the American Statistical Association, 66(336), 1971, 846-850.
- [17] Storn R., Price K., *Differential Evolution – A simple and efficient adaptive scheme for global optimisation over continuous spaces*, raport instytutowy TR-95-012, ICSI, 1995.
- [18] Storn R., *On the Usage of Differential Evolution for Function Optimization*, NAFIPS'96, 1996, 519-523.
- [19] Storn R., Price K., *Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces*, Journal of Global Optimization 11, 1997, 341-359.
- [20] Storn R., *Differential Evolution Homepage*, [online] 2010-11-30, (dostępny w Internecie: <http://www.icsi.berkeley.edu/~storn/code.html>).
- [21] Wagstaff K., Cardie C., *Clustering with Instance-level Constraints*, Proceedings of the 17th International Conference on Machine Learning, 2000, 1103-1110.
- [22] Wagstaff K., Cardie C., Rogers S., Schroedl S., *Constrained k-means Clustering with Background Knowledge*, Proceedings of the 18th International Conference on Machine Learning, 2001, 577-584.