Chapter 7
Chemical Reaction-Based Optimization Algorithm for Solving Clustering Problems

Yugal Kumar, Neeraj Dahiya, Sanjay Malik, Geeta Yadav, and Vijendra Singh

7.1 Introduction

Clustering is an important data analysis and well-recognized technique in the field of data mining. Since past few decades, it has attained a wide awareness from research community having theoretical as well as practical views. It is an unsupervised classification technique, and there is no need for training and testing of data objects. This technique can split the data objects into different clusters using a distance criterion. Literature survey has revealed the use of this technique in diverse research domains [3, 4, 7, 8, 18, 19, 21] . In recent years, a number of heuristics algorithms have been reported to determine the optimal solution for clustering problems. These algorithms have been derived from the heuristics, natural phenomena, swarms behavior, insects, animals, etc. Some physics-based algorithms (such as charged system search algorithm, galaxy-based algorithm, black hole optimization algorithm, magnetic charged system search algorithm) have also been used to find out the optimal solution for optimization problems [6, 9, 10, 16]. Some other algorithms like CSS [11], MCSS [12], BH [6], and CSO [13, 14] have also been applied successfully to solve the clustering problems. With time, the

Y. Kumar (✉)
Department of Computer Science and Engineering, JUIT, Waknaghat, Himachal Pradesh, India

N. Dahiya · S. Malik
Department of Computer Science and Engineering, SRM University, Delhi-NCR Campus, Ghaziabad, Uttar Pradesh, India

G. Yadav
Department of Pharmacy, Manav Bharti University, Solan, Himachal Pradesh, India

V. Singh
Department of Computer Science and Engineering, The Northcap University, Gurugram, India

X. Li, K.-C. Wong (eds.), *Natural Computing for Unsupervised Learning*,
Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-319-98566-4_7

importance and significance of these algorithms have increased among the research community due to following reasons:

- These algorithms have a variety of solution strategies.
- These algorithms do not depend on the size of the problem, variables, constraints, and solution space.
- These algorithms can be adopted according to the problem. Only need to adjust the objectives and fitness functions. It means no hard mathematical functions are defined in these algorithms.
- These algorithms provide efficient and effective results for combinatorial and nonlinear problems.
- These algorithms have the low computation power and minimum chances to trap in local optima.
- These algorithms have the capacity to solve the problem with different initial points.
- These algorithms provide better results as compared to the classical methods.

Due to abovementioned advantages, these algorithms have been used in diverse fields such as image processing, biology, sciences, market research, biomedical process, engineering, etc. Some new versions of these algorithms also have been proposed. Artificial chemical reaction optimization (ACRO) algorithm is a chemistry-based heuristic algorithm to solve the global optimization problems [1, 2]. This algorithm is based on the chemical reactions such as their types, evaluation, decomposition, etc. ACRO algorithm is mainly described in terms of reactants. For reactants, any encoding schemes, like integer, real, floating, and character, are used for solving the optimization problems. Further, in the ACRO algorithm, some reaction rules are specified to consume and generate the reactants. The execution of the algorithm is stopped when termination condition is met. The primary objective of this work is to investigate the competences of the ACRO algorithm for finding the optimal cluster centers for partitional clustering problems. The performance of proposed algorithm is examined on well-known benchmark datasets downloaded from UCI repository and few artificial datasets. The simulation results are compared with several meta-heuristic algorithms such as K-means [15], particle swarm optimization (PSO) [20], ant colony optimization (ACO) [17], and bat algorithm (BA) [5].

Organization of the Paper: The ACRO algorithm and different chemical reactions adopted in the algorithm are presented in Sect. 7.2. Section 7.3 presents the ACRO algorithm for partitional clustering problems. Section 7.4 illustrates the results of the proposed study. Section 7.5 gives a summary of the entire work.

7.2 Chemical Reactions

The chemical reaction can be defined as the formation of new chemical products using chemical reactants. Reaction completion timings may differ like some chemical reactions take few minutes, while others may take hours or days. Some

chemical reactions require single step for product formation, while others need multiple steps. In general, chemical reactions can be divided into two categories such as consecutive reaction and competing reaction. Basically, the motion of the electrons is generally responsible for the formation and breakage of the chemical bond in chemical reactions. It is also possible that the newly derive products, may differ in properties from reactants. A chemical process contains different types of molecules or chemical species, and further, these molecules can participate in one or more chemical reactions. Reactions may be exothermic or endothermic in nature. Reactions may be reversible or irreversible. In some reactions, the rate of the forward reaction is equal to rate of the reverse reaction such condition is known as chemical equilibrium condition. In equilibrium state, concentrations of reactants and products are constant. Observation shows that sometime output of a reaction is the input of other reaction. Broadly, chemical reactions can be characterized into following categories:

7.2.1 Synthesis Reactions

In these reactions, two or more reactants combine for producing a single product or compound. The reaction is given below.

$$2K + Cl_2 \rightarrow 2KCl \qquad \text{(Elements)}$$

$$2KCl + 3O_2 \rightarrow 2KClO_3 \qquad \text{(Compounds)}$$

7.2.2 Decomposition Reactions

Through this reaction, two or more compounds form using a single reactant. It just reverses of the synthesis reaction. Decomposition reactions need some energy source like heat, light, or electricity. A decomposition reaction can be shown as

$$MgCl_2 \rightarrow Mg + Cl_2$$

7.2.3 Single Displacement Reactions

In single displacement reaction, an element can interact with another element of a compound and takes position of another element in the resultant compound. This reaction can be given as

$$Zn + 2HCl \rightarrow 2ZnCl_2 + H_2$$

7.2.4 Metathesis

In metathesis reactions, after the interaction of the elements, the position of cations and anions changes and produce two new elements. This reaction can be specified as given below.

$$AgNO_3 + NaCl \rightarrow 2AgCl + NaNO_3$$

7.2.5 Combustion Reactions

In this reaction, oxygen reacts with an element and produced energy in terms of heat and light. Combustion reactions can be given as

$$C_2H_5 + 3O_2 \rightarrow 2CO_2 + 3H_2O$$

7.2.6 Redox Reactions

In this reaction, electrons transfer from one reactant to another reactant in terms of release and uptake. The element that can uptake the electrons is called an oxidizing agent, whereas the element that can release the electrons is known as the reducing agent. The example of such reaction is given as

$$MnO_4^- (aq) + I^- (aq) \rightarrow Mn^{+2} (aq) + I_2 (s)$$

Mn gains five electrons to form Mn^{2+}. Further, iodine lost electrons, so it is an oxidizing agent. Moreover, Mn^{2+} gains five electrons, so it is a reducing agent.

7.2.7 Reversible Reactions

These reactions occur both in forward and backward direction. In such types of reactions, the products change into reactants and reactants into products. The example of such reaction is given below.

$$H_2CO_3 (l) + H_2O (l) \leftrightarrow \ HCO_3{}^- (aq) + H_3O (aq)$$

7.3 Artificial Chemical Reaction Optimization (ACRO) Algorithm

The motivation behind the artificial chemical reaction optimization (ACRO) algorithm is the chemical reaction. It is discovered that atoms and molecules constantly move and collide in a viscous fluid filled 2D cellular space. Atoms are defined as the elementary particle of a chemical reaction and having a type, charge, mass, position, radius, orientation, and velocity. Whereas, a molecule can be defined as a group of two or more atoms linked through bonds. During a chemical reaction, several changes occur like formation and breaking of bonds, orientation change, substitution, and replacement in molecules. It is considered that ACRO algorithm begins with the action of reactants in a vessel. Let us consider a fixed volume vessel filled with a uniform mixture of N chemical reactants which undergo different types of chemical reactions. Let $Ri \in (1 \leq i \geq N)$ is the list of chemical types, and it is considered that these types can cooperate through M specified chemical reaction channels. In ACRO algorithm, the encoding scheme of the reactants depends on the user choice, and it can be binary, real, string, integer, etc. The new reactant produces through the interaction of one or two reactants. The ACRO algorithm starts with a set of initial reactants in a solution. Further, reactants are consumed and produced through different chemical reactions. The algorithm stops; either the termination condition is met or when no more reaction occurs. For the chemical reactions, reactants are chosen based on their concentrations and potentials. Moreover, it is observed that consecutive and competing reactions are the two mainly reported reactions. Reactants are joined together serially in a consecutive reaction, whereas in competing type reactions, different products are produced depending on the specific condition. It is also noticed that sometimes output of one reaction may act as a reactant for other reaction. There are many factors that can affect the execution of a reaction. However, the ACRO algorithm is based on a simple concept of an equal probability of monomolecular or bimolecular reactions and their alternatives. The main steps of the ACRO algorithm are listed below.

Step 1: Initialize the problem and algorithm parameters.
Step 2: Evaluate the initial reactants and evaluation.
Step 3: Apply chemical reactions.
Step 4: Update reactants.
Step 5: Termination condition.

7.4 Proposed ACRO Algorithm for Clustering Problems

This section describes the proposed ACRO algorithm to find the optimal cluster centers for the partition-based clustering problems. In partition-based clustering problems, numbers of clusters are known in advance, and the main task is to find the

optimum set of cluster centers such that the intra-cluster distance between cluster centers and objects should be minimized. The steps of the proposed algorithm are described in Sect. 7.4.1.

7.4.1 Steps of Proposed ACRO Algorithm

Step 1: Initialization of the User-Defined Parameters and Problem Statement
The clustering problem can be expressed as minimization problem and defined as follows:

$$\text{Minimize } f\left(\boldsymbol{x} \textbf{ subject } \boldsymbol{to}\ \boldsymbol{x_i} \in \boldsymbol{C_k}; \textbf{ where } \boldsymbol{i} = \mathbf{1}, \mathbf{2}, \mathbf{3}, \ldots, \boldsymbol{D} \textbf{ and } \boldsymbol{k} = \mathbf{1}, \mathbf{2}, \ldots, \boldsymbol{K}\right)$$

where $f(x)$ is an objective function which is defined in terms of Euclidean distance, $x_i$is the set of data objects, D is the dimension of data object, C_k is the set of k^{th} cluster centers, and K describes a number of clusters in a given dataset. It is to be noted that the computed cluster centers should be in the range of $x_i^{\min}$and $x_i^{\max}$and also known as boundary constraints. Further, in this work, integer encoding scheme is used to obtain the desired results. The user-defined parameters of ACROA and Reactnum (maximum number of iterations) are also defined in this step.

Step 2: Set Initial Reactants and Evaluation
In this step, the population of the ACRO algorithm is specified; the population is defined in terms of initial reactants. The initial reactants are uniformly identified from the feasible search space. For clustering problem, initially, two reactants, i.e., R_1 and R_2, are identified from the dataset such as $R_1 = \{x_{i,1}, x_{i,2}, \ldots, x_{i,d}\}$, $R_2 = \{x_{j,1}, x_{j,2}, \ldots, x_{j,d}\}$, where d denotes the length (dimension) of reactant. The number of reactants is similar to a number of clusters (K). Then, rests of reactants are derived from initially determined reactants R_1 and R_2 using the following procedure. To compute R_1 and R_2, firstly, a dividing factor (k) is initialized. Suppose $k = 2$; then, two new extra reactants, i.e., R_3, R_4, are generated from R_1 and R_2 using below mentioned procedure:

$$R_3 = \left\{ \begin{array}{c} r * x_{i,1}, r * x_{i,2}, \ldots, r * x_{i,\frac{d}{2}}; \\ r * x_{j,\frac{d}{2}+1}, r * x_{j,d-\frac{1}{2}+2}, \ldots, r * x_{j,1} \end{array} \right\} \tag{7.1}$$

$$R_4 = \left\{ \begin{array}{c} r * x_{j,1}, r * x_{j,2}, \ldots, r * x_{j,\frac{d}{2}}; \\ r * x_{i,\frac{d}{2}+1}, r * x_{i,\ d-\frac{1}{2}+2}, \ldots, r * x_{i,1} \end{array} \right\} \tag{7.2}$$

where r denotes a random number and should be in the range of $0 \leq r \geq 1$. Further, more reactants are generated using the following procedure until the number of reactants is not similar to the desired clusters (K):

$$R_5 = \left\{ \begin{array}{c} r * x_{i,1}, r * x_{i,2}, \ldots, r * x_{i,\frac{2d}{3}}; \\ r * x_{j,2d/3+1}, r * x_{j,2(d-1)/3+2}, \ldots, r * x_{j,1} \end{array} \right\} \tag{7.3}$$

$$R_6 = \left\{ \begin{array}{c} r * x_{i,1}, r * x_{i,2}, \ldots, r * x_{i,\frac{d}{3}}; \quad r * x_{j,d/3+1}, r * x_{j,2(d-1)/3+2}, \\ r * x_{i,\frac{2d}{3}}, \ldots, r * x_{j,1} \end{array} \right\} \tag{7.4}$$

$$R_7 = \left\{ r * x_{i,1}, r * x_{i,2}, \ldots, r * x_{i,\frac{d}{3}}; \quad r * x_{j,d/3+1}, , \ldots, r * x_{j,1} \right\} \tag{7.5}$$

$$R_8 = \left\{ r * x_{j,1}, r * x_{j,2}, \ldots, r * x_{j,\frac{d}{3}}; \quad r * x_{i,\frac{d}{3}}, \quad \ldots, r * x_{i,1} \right\} \tag{7.6}$$

$$R_9 = \left\{ \begin{array}{c} r * x_{j,1}, r * x_{j,2}, \ldots, r * x_{j,\frac{d}{3}}; \quad r * x_{i,\frac{d}{3}+1}, r * x_{i,\frac{2(d-1)}{3}}, \\ r \ * x_{j,2d/3+1} \ldots, r * x_{i,1} \end{array} \right\} \tag{7.7}$$

$$R_{10} = \left\{ r * x_{j,1}, r * x_{j,2}, \ldots, r * x_{j,\frac{2d}{3}}; r * x_{i,\frac{2(d-1)}{3}+1}, r * x_{i,\frac{2d}{3}+2}, \ldots, r * x_{i,1} \right\} \tag{7.8}$$

Step 3: Applying Chemical Reactions

Step 3.1: Bimolecular Reactions

Let us consider $R_1 = \{x_{i,1}, x_{i,2}, \ldots, x_{i,d}\}$ and $R_2 = \{x_{j,1}, x_{j,2}, \ldots, x_{j,d}\}$ are two reactants that can participate in bimolecular reaction. As a result of this, various bimolecular reactions are incorporated in the ACRO algorithm such as synthesis reaction, displacement reaction, redox 2 reaction, and monomolecular reaction. For all these operations, integer representation of the population is considered rather than binary encoding. The detailed explanation of these reactions is described as below.

Step 3.2: Synthesis Reaction: Using this reaction, a new reactant can be obtained using the following equation:

$$R = \left(r_1, r_2, r_3, \ldots, r_i, r_j, r_k, \ldots, r_n\right); \text{where, } R_{i,\text{new}} = R_i + \lambda_i \left(R_j - R_i\right) \tag{7.9}$$

In the above equation, λ_i is a random value in the interval [0.25, 1.25]; R_i and R_j are randomly selected reactants.

Step 3.3: Displacement Reaction

The new reactants are obtained using below mentioned procedure. Suppose $R_k = (R_1, R_2 \ldots R_i, R_j, \ldots, R_K)$ where, $k = 1,2 \ldots \ldots .K$:

$$\mathrm{R_{i,new}} = R_i \left(1 - \lambda_{\mathrm{td}} R_j\right) \tag{7.10}$$

$$R_{j,\text{new}} = \lambda_{\mathrm{td}} R_j + (1 - \lambda_{\mathrm{td}} R_i) \tag{7.11}$$

where $\lambda_{td} \in \{0, 1\}$ and$\lambda_{td+1} = 2.3\ (\lambda_{td})^{2\sin(\pi\lambda_{td})}$ and where in λ_{td} the suffix "td" is increased by 1, when the reaction is completed.

Step 3.4: Redox 2 Reaction

In this reaction, if R_1 is a better reactant in terms of the objective function, then in $\lambda_{tr,}$ the suffix "tr" is updated by using 1 when the reaction is performed and it is computed using the following equation:

$$R_{i,\text{new}} = R_i + \lambda_{tr}\left(R_i - R_j\right); \text{where,}\quad \lambda_{tr} \in \{0, 1\} \tag{7.12}$$

Otherwise, if R_2 is a better reactant, then in $\lambda_{tr,}$ the suffix "tr" is updated by using 1 when the reaction is performed and it is computed using the following equation:

$$R_{j,\text{new}} = R_j + \lambda_{tr}\left(R_j - R_i\right); \text{where,}\quad \lambda_{tr} \in \{0, 1\} \tag{7.13}$$

The value of λ_{tr} is computed using the following equation:

$$\lambda_{tr+1} = \begin{cases} 0 & \lambda_{tr} = 0 \\ \frac{1}{\lambda_{tr} \bmod(1)}\lambda_{tr} & \lambda_{tr} \in (0, 1) \end{cases} \tag{7.14}$$

$$1/\lambda_{tr} \bmod(1) = \frac{1}{\lambda_{tr}} - \left\lfloor \frac{1}{\lambda_{tr}} \right\rfloor \tag{7.15}$$

Step 3.4: Monomolecular Reactions

Step 3.4.1: Decomposition Reaction

In this reaction, suppose $R = (R_1, R_2 \ldots R_i, R_j, \ldots, R_n)$ is the reactant and $C_k \in \{R_m, R_n\}$ is an atom that takes part in monomolecular reaction. The new atom of the molecule $R_{i,\text{ new}}$ is a random population or reactant from $\{R_m, R_n\} \in C_k$, but $\{R_m, R_n\} \neq (R_i \text{ and } R_j)$.

Step 3.4.2: Redox 1 Reaction

In this reaction, the new reactant is generated using the following procedure:

$$R_{i,\text{new}} = R_m + \lambda_t\ (R_n - R_m) \tag{7.16}$$

where $\lambda_t \in \{0, 1\}$ such that initial $\lambda_0 \in \{0.0, 0.25, 0.5, 0.75, 1.0\}$ and $\lambda_{t+1} = 4\lambda_t(1 - \lambda_t)$; t is updated by using 1 when the reaction is performed.

Step 4: Reactants Update

In this step, a chemical equilibrium test is carried out. If the fitness function of the new generated reactants is better than another, the generated reactant includes into the chemical reaction process and excludes the worst reactant from reversible chemical reactions.

Table 7.1 Description of the datasets used for experiment

Dataset	Cluster (K)	Attributes	Objects	Objects in clusters
Iris	3	4	150	(50, 50, 50)
Wine	3	13	178	(59, 71, 48)
CMC	3	9	1473	(629,334, 510)
Cancer	2	9	683	(444, 239)
Glass	6	9	214	(70,17, 76, 13, 9, 29)

Step 5: Termination Condition
If the Reactnum is equal to the user-defined maximum number of iterations, then the algorithm stops its execution and produces the optimal cluster centers; otherwise steps 3 and 4 are repeated, until the desired results are not obtained or termination condition is not satisfied.

7.5 Simulation Results

This section presents the experimental results of our study. To examine the efficacy of the proposed ACRO based clustering algorithm, some well-known benchmark datasets are taken from the UCI repository. These datasets are iris, CMC, glass, wine, and cancer. The description these datasets are reported in Table 7.1. Further, in this work, intra-cluster distance and f-measure parameters are adopted as performance measure parameters. The intra-cluster distance parameter presents the quality of clusters, and it can be measured using best, average, and worst intra-cluster distances. The precision can be computed using f-measure parameter. The MATLAB environment is used to implement the proposed algorithm using Windows-based operating system. Table 7.1 demonstrates the details of datasets taken for the experiments. The simulation results of proposed algorithm are compared with other well-known clustering algorithms K-means [15], particle swarm optimization (PSO) [20], ant colony optimization (ACO) [17], and bat algorithm (BA) [5].

7.5.1 Performance Matrices

In this section, different performance matrices are described that are adopted to compute the performance of the ACRO algorithm for partitional clustering problems. In this work, f-measure and intra-cluster distance parameters are taken to evaluate the performance of the algorithms. These matrices are explained as below.

- Intra-cluster distances

This parameter is used to compute the distance between data objects and cluster centers and also represents the quality of clustering. The results describe using best, average, and worst cases.

- F-Measure

It is expressed in terms of recall and precision. The value of f-measure parameter is computed through Eq. 7.17.

$$F\left(i,j\right) = \sum_{i=1}^{n} \frac{n_i}{n}^{*} \max_{i}^{*} F\left(i,j\right) \tag{7.17}$$

Where, $F(i,j)$is determined using Eq. 7.18.

$$F\left(i,j\right) = \frac{2 * (\text{Recall} * \text{Precision})}{(\text{Recall} + \text{Precision})} \tag{7.18}$$

The recall and precision are measured using Eq. 7.19.

$$\text{Recall}\left(r\left(i,j\right)\right) = \frac{n_{i,j}}{n_i} \quad \text{and} \quad \text{Precision}\left(p\left(i,j\right)\right) = \frac{n_{i,j}}{n_j} \tag{7.19}$$

7.5.2 Results and Discussion

This subsection describes the results of our study using proposed algorithm. Tables 7.2 and 7.3 present the results of the proposed ACRO algorithm and other clustering using artificial dataset 1 and artificial dataset 2. It is noticed that the proposed algorithm gives better results in comparison to all other algorithms. It is also observed that K-means algorithm obtains worst results among all other algorithm using intra-cluster distance parameter both of artificial dataset1 and artificial dataset 2 using intra-cluster distance parameter.

Tables 7.4 and 7.5 illustrate the results of the proposed algorithm and other algorithms using iris and cancer dataset. It is reported that the proposed algorithm

Table 7.2 Performance comparison of proposed ACRO and other algorithms using artificial dataset 1

Parameters	K-means	PSO	ACO	CSO	BA	Proposed ACRO
Best case	161.32	153.45	154.29	159.73	151.81	143.36
Avg. case	166.12	161.24	158.52	163.37	154.98	149.56
Worst case	174.64	169.39	165.42	168.13	159.08	154.21
SD	7.625	6.437	4.712	5.418	4.564	5.243
F-Measure	0.94	0.96	0.99	0.95	0.97	1

Table 7.3 Performance comparison of proposed ACRO and other algorithms using artificial dataset 2

Parameters	K-means	PSO	ACO	CSO	Bat	Proposed ACRO
Best case	761.45	753.24	759.87	752.34	756.09	746.53
Avg. case	768.38	759.82	766.15	756.21	761.44	752.26
Worst case	776.49	769.58	773.41	764.39	169.17	758.13
SD	6.837	7.614	6.845	4.936	5.432	4.214
F-Measure	0.89	0.91	0.93	0.937	0.946	0.976

Table 7.4 Performance comparison of proposed ACRO and other algorithms using iris dataset

Parameters	K-means	PSO	ACO	CSO	Bat	Proposed ACRO
Best case	97.52	97.05	97.21	96.98	96.84	95.56
Avg. case	113.56	98.73	98.36	97.64	97.53	96.73
Worst case	125.23	99.89	99.59	98.78	98.09	97.48
SD	15.326	0.467	0.426	0.392	0.263	0.196
F-Measure	0.781	0.78	0.778	0.781	0.782	0.785

Table 7.5 Performance comparison of proposed ACRO and other algorithms using cancer dataset

Parameters	K-means	PSO	ACO	CSO	Bat	Proposed ACRO
Best case	2989.46	2978.68	2983.49	2985.16	2972.36	2912.66
Avg. case	3248.25	3116.64	3178.09	3124.15	3098.93	3063.34
Worst case	3566.94	3358.43	3292.41	3443.56	3282.75	3179.25
SD	256.58	107.14	93.45	128.46	56.24	71.22
F-Measure	0.832	0.826	0.829	0.831	0.833	0.835

obtains better quality results than other algorithms. Further, it is noted that the K-means algorithm obtains a maximum intra-cluster distance among all algorithms using both of datasets. It is also noticed that the performance of the K-means, PSO, ACO, and CSO algorithm is similar in case of f-measure parameter using iris dataset. But, the significant difference occurs in terms of intra-cluster distance parameter. It is observed that the PSO algorithm has worst performance using f-measure parameter among the rest of algorithms for cancer dataset.

Tables 7.6, 7.7, and 7.8 demonstrate the results of proposed ACRO and other algorithms using CMC, wine, and glass dataset. It is revealed that the proposed algorithm provides enhanced results in comparison to other algorithms for CMC and wine datasets. Moreover, it is noticed that the performance of BA algorithm is better for the glass dataset in comparison to all other algorithms. Further, it is reported that K-means algorithm obtains maximum intra-cluster distance for CMC and wine datasets, whereas, an ACO algorithm obtains a maximum intra-cluster distance for glass dataset. For wine and glass datasets, K-means and ACO algorithms exhibit worst f-measure results. In case of the CMC dataset, PSO, ACO, and CSO provide similar f-measure results.

Table 7.6 Performance comparison of proposed ACRO and other algorithms using CMC dataset

Parameters	K-means	PSO	ACO	CSO	Bat	Proposed ACRO
Best case	5828.25	5792.48	5756.42	5712.78	5689.16	5681.56
Avg. case	5903.82	5846.63	5831.25	5804.52	5778.14	5746.32
Worst case	5974.46	5936.14	5929.36	5921.28	5914.25	5894.63
SD	49.62	48.86	44.34	43.29	39.54	36.41
F-Measure	0.337	0.333	0.332	0.334	0.336	0.339

Table 7.7 Performance comparison of proposed ACRO and other algorithms using wine dataset

Parameters	K-means	PSO	ACO	CSO	Bat	Proposed ACRO
Best case	16768.18	16483.61	16448.35	16431.76	16372.02	16256.42
Avg. case	18061.24	16417.47	16530.53	16395.18	16357.89	16336.21
Worst case	18764.49	16594.26	16616.36	16589.54	16556.76	16396. 56
SD	796.13	88.27	48.86	62.41	41.78	37.83
F-Measure	0.519	0.516	0.522	0.521	0.523	0.526

Table 7.8 Performance comparison of proposed ACRO and other algorithms using glass dataset

Parameters	K-means	PSO	ACO	CSO	Bat	Proposed ACRO
Best case	222.43	264.56	273.22	256.53	256.47	261.47
Avg. case	246.51	278.71	281.46	264.44	261.61	266.23
Worst case	258.38	283.52	286.08	282.27	278.24	274.14
SD	18.32	8.59	6.58	15.43	7.09	8.11
F-Measure	0.426	0.412	0.402	0.416	0.430	0.428

Figures 7.1, 7.2 and 7.3 illustrate the dispersion of the data objects in artificial dataset 1 and dataset 2, respectively. These artificial datasets are generated in the MATLAB. Artificial dataset 1 is a two-dimensional dataset, whereas artificial dataset 2 is three-dimensional datasets. Figures 7.2, 7.3 and 7.4 show the clustering of the data objects into different clusters using the proposed algorithm.

Figures 7.5 and 7.6 show the clustering of the iris dataset using proposed artificial chemical reaction optimization algorithm. The data objects of iris dataset are divided into three clusters such as setosa, versicolor, and virginica. It is seen that data objects in setosa cluster are linearly separable from versicolor and virginica, while the data objects of versicolor and virginica clusters are linearly inseparable that can also affect the performance of the algorithm. But, it is observed that the proposed algorithm gives better results for all three clusters. Figure 7.5 shows the illustration of the data objects of the iris dataset using sepal width and petal width attributes, whereas Fig. 7.6 shows the illustration of data objects using petal length, sepal width, and petal width.

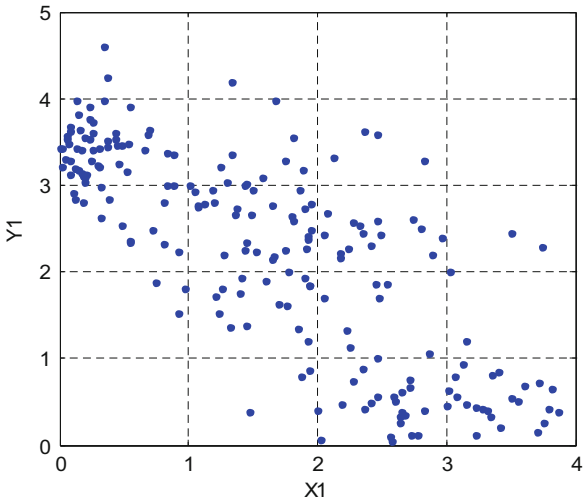

Fig. 7.1 Distribution of data objects in artificial dataset 1

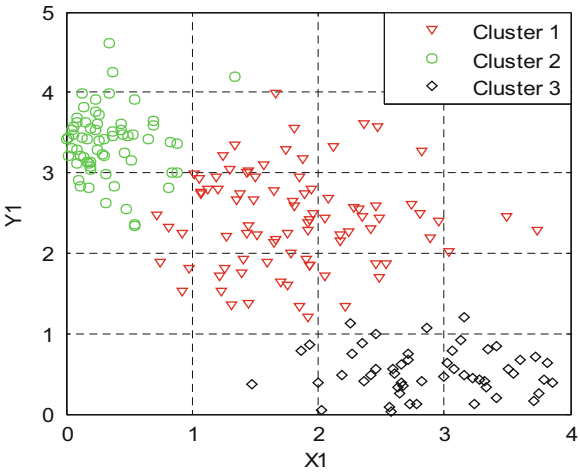

Fig. 7.2 Clustering results of proposed ACRO algorithm on artificial dataset 1

7.6 Conclusion

In this work, an artificial chemical reaction optimization algorithm is presented for solving partitional clustering problems. The proposed algorithm is inspired by the chemical reaction process. In this algorithm, reactants are used to search the optimal solution, and these reactants are uniformly determined from the search space. Further, optimal solution for the problems can be represented using the reactants. The main work of the ACRO algorithm is to measure optimal cluster centroid for partitional clustering problems. The reactants represent the initial cluster centers, which are determined uniformly from the dataset. The proposed algorithm is applied to optimize the value of initially chosen reactants through its

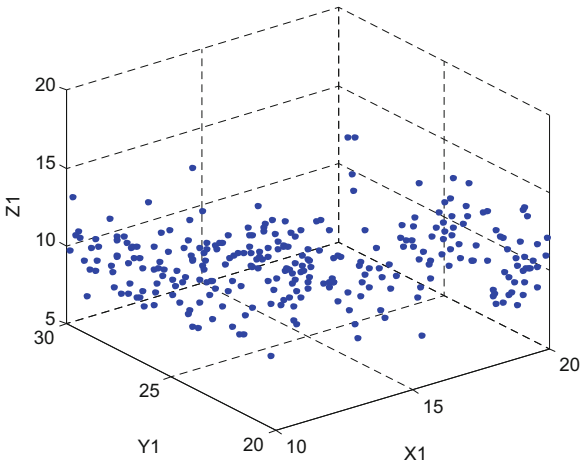

Fig. 7.3 Distribution of data objects in artificial dataset 2

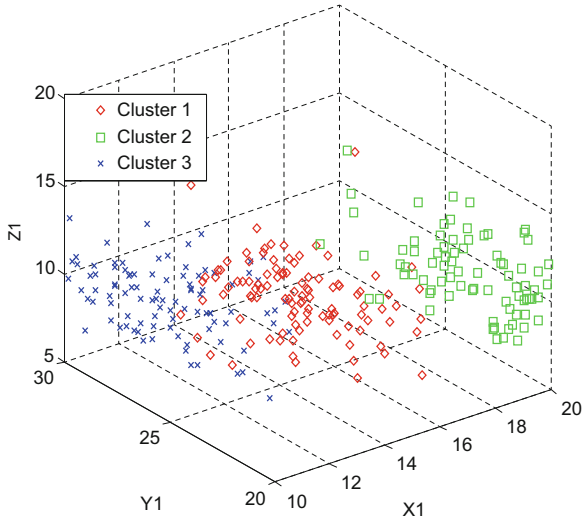

Fig. 7.4 Clustering results of proposed ACRO algorithm on artificial dataset 2

various steps. The performance of the proposed algorithm is tested on several real-life clustering problems and compared with state-of-the-art clustering algorithms. From the simulation results, it is noticed that the proposed algorithm achieves better clustering results in comparison to other clustering algorithms. Finally, it is stated that proposed ACRO algorithm is one of the efficient and effective algorithm for solving partitional clustering problems.

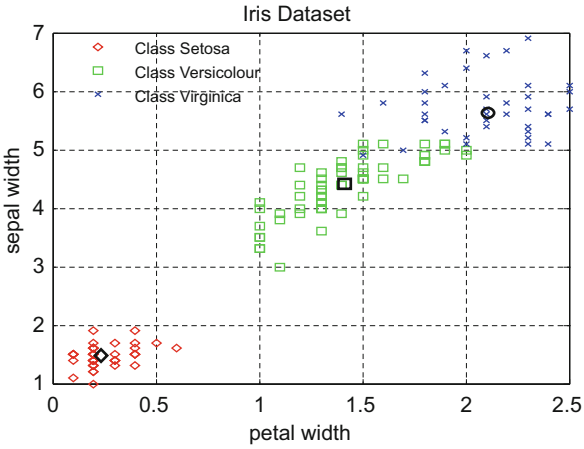

Fig. 7.5 Clustering results of proposed ACRO algorithm on iris dataset (2D view)

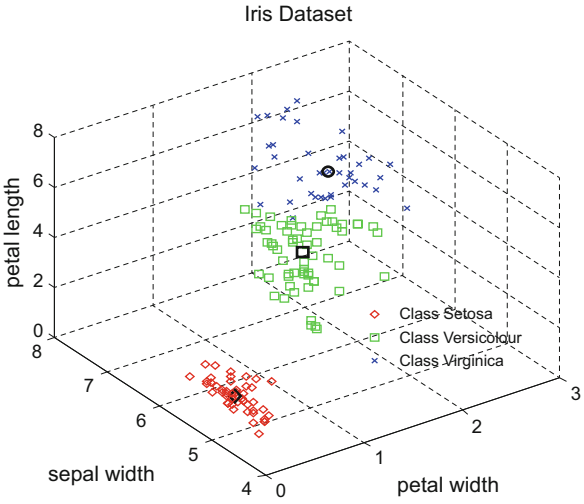

Fig. 7.6 Clustering results of proposed ACRO algorithm on iris dataset (3D view)

References

1. Alatas B (2011) ACROA: artificial chemical reaction optimization algorithm for global optimization. Expert Syst Appl 38(10):13170–13180
2. Alatas B (2012) A novel chemistry based metaheuristic optimization method for mining of classification rules. Expert Syst Appl 39(12):11080–11088
3. Anaya AR, Boticario JG (2011) Application of machine learning techniques to analyses student interactions and improve the collaboration process. Expert Syst Appl 38(2):1171–1181
4. Dunn WJ III, Greenberg MJ, Callejas SS (1976) Use of cluster analysis in the development of structure-activity relations for antitumor triazenes. J Med Chem 19(11):1299–1301
5. Gandomi AH, Yang XS, Alavi AH, Talatahari S (2013) Bat algorithm for constrained optimization tasks. Neural Comput Applic 22(6):1239–1255
6. Hatamlou A (2013) Black hole: a new heuristic optimization approach for data clustering. Inform Sci 222:175–184

7. He Y, Pan W, Lin J (2006) Cluster analysis using multivariate normal mixture models to detect differential gene expression with microarray data. Comput Stat Data Anal 51(2):641–658
8. Hung YS, Chen KLB, Yang CT, Deng GF (2013) Web usage mining for analyzing elder self-care behavior patterns. Expert Syst Appl 40(2):775–783
9. Kaveh A, Talatahari S (2010) A novel heuristic optimization method: charged system search. Acta Mech 213(3):267–289
10. Kaveh A, Share MAM, Moslehi M (2013) Magnetic charged system search: a new meta-heuristic algorithm for optimization. Acta Mech 224(1):85–107
11. Kumar Y, Sahoo G (2014) A charged system search approach for data clustering. Prog Artif Intell 2(2-3):153–166
12. Kumar Y, Sahoo G (2015) Hybridization of magnetic charge system search and particle swarm optimization for efficient data clustering using neighborhood search strategy. Soft Comput 19(12):3621–3645
13. Kumar Y, Sahoo G (2015) A hybrid data clustering approach based on improved cat swarm optimization and K-harmonic mean algorithm. AI Commun 28(4):751–764
14. Kumar Y, Sahoo G (2016) A hybridize approach for data clustering based on cat swarm optimization. Int. J Inf Commun Technol 9(1):117–141
15. MacQueen J (1967) Some methods for classification and analysis of multivariate observations. In: Fifth Berkeley Symposium on Mathematics. Statistics and Probability. University of California Press, pp 281–297
16. Shah-Hosseini H (2011) Principal components analysis by the galaxy-based search algorithm: a novel metaheuristic for continuous optimisation. Int J Comput Sci Eng 6(1-2):132–140
17. Shelokar PS, Jayaraman VK, Kulkarni BD (2004) An ant colony approach for clustering. Analy Chim Acta 509(2):187–195
18. Teppola P, Mujunen SP, Minkkinen P (1999) Adaptive Fuzzy C-Means clustering in process monitoring. Chemom Intel Lab Syst 45(1):23–38
19. Webb A (2002) Statistical pattern recognition. Wiley, New Jersey, pp 361–406
20. Zhan ZH, Zhang J, LiY CSH (2009) Adaptive particle swarm optimization. IEEE Trans Syst Man Cybern B Cybern 39:1362–1381
21. Zhou H, Liu Y (2008) Accurate integration of multi-view range images using k-means clustering. Pattern Recognit 41(1):152–175