# An Improved Density Peak Clustering Algorithm Based on Gravity Peak

Hui Han[1] and Rui Zhang[2(✉)]

[1] State Key Laboratory of Astronautic Dynamics, Xi'an 710043, Shaanxi, China
[2] School of Information Engineering, Shandong Management University, No. 3500, Dingxiang Road, Changqing District, Jinan 250357, Shandong, China
14438120201031@sdmu.edu.cn

**Abstract.** Density peak clustering (DPC) algorithm has been cited and further improved by many researchers since it was put forward. In the aspect of cluster centers discovery, the locations of these points need manually judged by classical decision graph or improved decision graph. However, this way of manual participation in decision-making significantly reduces the efficiency of the algorithm and trades the cost of efficiency for accuracy. Although, relevant scholars have made some improvements on the decision graph to automatically determine the truncation distance or cluster center. To solve this problem, an improved density peak algorithm based on gravitational peak named IDPC-GP is proposed. In this approach, the gravitational dimension is introduced into the data space in order to better grasp the data distribution, and KNN similarity is used for extended clustering. Then, the cluster centers can be quickly found and clustered. Experiments verifies the superiority of the algorithm in comprehensive performance of the IDPC-GP algorithm.

**Keywords:** Density peak clustering · Gravity center · KNN

## 1 Introduction

As a new generation of high-performance density clustering algorithm, density peak clustering algorithm provides a new starting point and research direction for unsupervised clustering analysis [1, 2]. DPC algorithm has superior performance in discovering high-density centers [3], special-shaped clusters and processing unbalanced data. However, the high-density center of DPC is determined manually based on the decision graph. And whether the high-density center can be clearly distinguished depends on the setting of appropriate truncation radius parameters [4]. To solve the above problems, relevant scholars have made relevant improvements to improve the adaptability and parameter tolerance of DPC algorithm. Li Tao et al. user $\gamma$ ranking graph determines the inflection point and potential cluster center, and then automatically determines the actual cluster center from the potential cluster center [5]. The effect of self-determination center has been realized. Jiang P et al. used the improved adaptive method to select the representative points of the core grid as the clustering center to solve the problem that the


Y. Wang et al. (Eds.): ICSINC 2022, LNEE 996, pp. 284–291, 2023.
https://doi.org/10.1007/978-981-19-9968-0_34

center cannot be determined adaptively [6]. Chen Jinyin et al. solved the problem that the density center is difficult to determine by fitting the density distance product density distribution with the normal distribution curve [7]. Ruhui Liu et al. proposed a constraint based fast density peak search clustering algorithm (CCFDP). Automatically generate multiple potential clustering centers, and make the information best of the structure in the constraints to determine the high-density centers [8]. In this paper, aiming to the problems of parameter adaptation and automatic determination of cluster center in DPC, an improved density peak clustering algorithm based on gravitational peak (IDPC-GP) is proposed.

## 2 Related Works

In this paper, the two problems to be solved: 1. The automatic determination of the center of DPC algorithm; 2. Reduce the definition of manual parameters and the tolerance rate of necessary parameters. To solve the first problem, two types of decision graphs are designed in DPC. The one is classical decision graph based on relative density distance $\delta$ and local density $\rho$ [9]; The other is the improved decision diagram obtained by multiplying and sorting the two parameter values, which is obtained by Eqs. (1), (2), (3) and (4).

$$\rho_i = \sum_{j=1}^{N} \chi\left(D_{ij} - D_c\right) \quad i, j \in [1, N] \tag{1}$$

$$\chi(\cdot) = \begin{cases} 1 \ \Delta D \leq 0 \\ 0 \ otherwise \end{cases} \tag{2}$$

$$\delta_i = \min\left(D_{ij}\right) \quad \rho_i < \rho_j \tag{3}$$

$$\gamma_i = \rho_i * \delta_i \tag{4}$$

where $N$ is the total amount of samples. $\triangle D = D_{ij} - D_c$. As to the second problem, the initialization of the algorithm needs to involve the truncation radius and the number of high-density centers. These two parameters need a priori knowledge on the one side and manual determination on the other hand. After adjusting the appropriate parameters, that is, when the high-density center is clearly distinguished from other sample points, take the high-density center as the starting point, and divide each sample point into the cluster where the nearest neighbor of high-density is located to complete the division. Take UCI data set *Jain* as an example.

It can be seen in Fig. 1 and 2 that when different $d_c$ values are taken, two high-density centers can be clearly seen in the classical and improved decision diagrams. But if the number is right, the locations of the high-density centers are not expected.

It the Fig. 3, when the $d_c$ value is inappropriate, the result is very poor, because the clustering is based on the center position. If the center position is wrong, it will inevitably lead to disastrous consequences of subsequent clustering.

In astrophysics, according to the current theory, gravity is a higher dimension than four-dimensional space-time, which has more advantages for grasping spatial distribution. The research based on data gravity comes from universal gravity, Lizhi Peng et al.

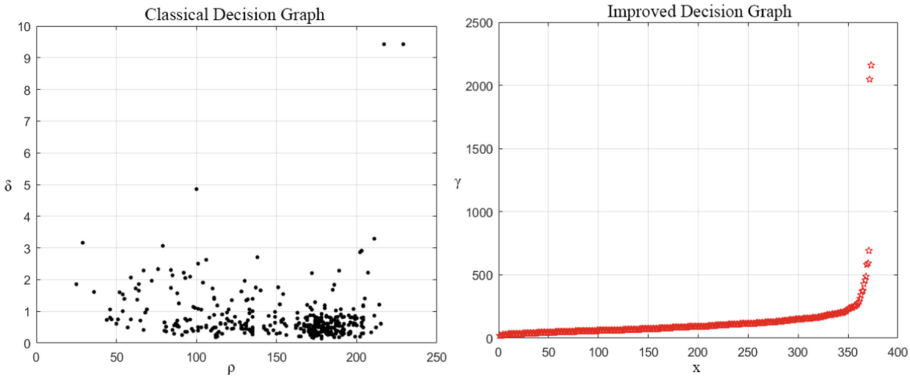


**Fig. 1.** Two kinds of decision graphs on Jain with $d_c = 10$

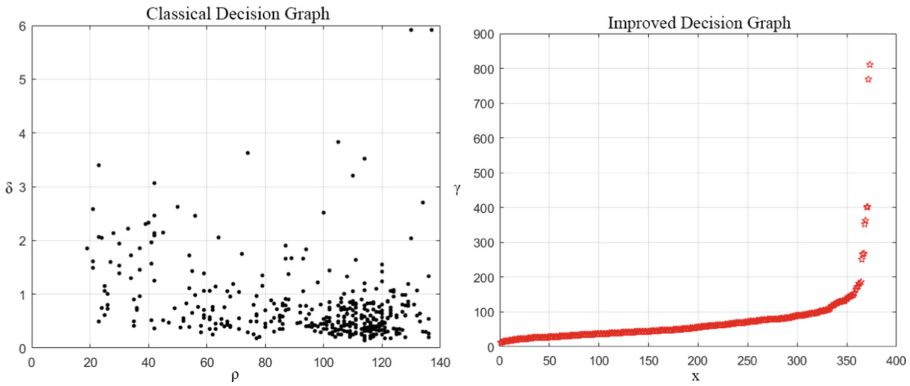


**Fig. 2.** Two kinds of decision graph on Jain with $d_c = 12$

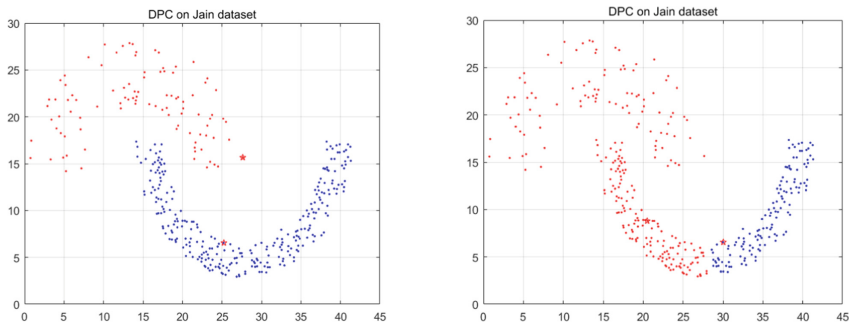


**Fig. 3.** Clustering results comparison on Jain between with $d_c = 10$ and $d_c = 12$

used data gravity for classification learning [10]. The gravity formula is shown in the Eq. (5).

$$F_{i,j} = G\frac{M_i * M_j}{r^2} \tag{5}$$

where $M_i$ and $M_j$ represents the mass of two celestial bodies and $r$ is gap between them. $G$ is a constant.

## 3 Method

DPC algorithm consists of two parts: the discovery of gravity center and extended clustering based on local spatial similarity. The algorithm will be designed from these two aspects.

### 3.1 Discovery of Gravity Centers

The data gravity center is related to the density of data distribution and local data volume. The judgment basis of the centers is based on the gravity value of k-nearest neighbor (KNN) of each sample. The data gravity formula evolved from Eq. (5), as shown in the Eq. (6).

$$F_i = \sum_{j=1}^{k} \frac{1}{r_{ij}^2} \qquad j \in [1, k] \tag{6}$$

Next, the gravity value between each sample and its k-nearest neighbors are compared. Those larger than the gravity value of all k-nearest neighbor samples are regarded as local gravity centers or potential gravity centers. Still take Jain data set as an example. When k is 10, the distribution diagram of local gravity center is obtained, as shown in the Fig. 4:

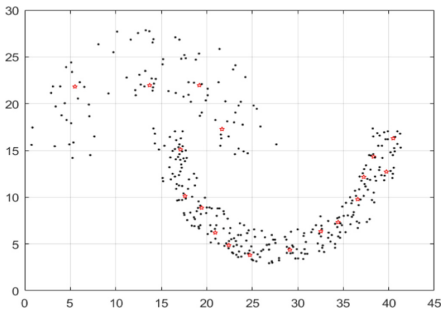


**Fig. 4.** Gravity center distribution (Red Star mark).

Obviously, there are two banded clusters in *Jain* dataset, and the gravity centers found are far more than 2. Fortunately, their distribution covers two clusters. What we

need to do next is not to screen the core cluster centers, because the decision graph has failed before. The first two high-density points obtained according to the $\gamma$ ranking graph are not the center points we want. DPC-GP algorithm does not directly consider the problem of further determining the center. Instead, it adjusts and filters in real time in the clustering process.

### 3.2 Clustering Process of IDPC-GP

After the centers are determined, cluster mining is started in an extended way, and classes are built by KNN similarity. All special cases are handled in the process of expansion, such as whether it is a cluster center in local high density. KNN similarity is defined as: suppose the set KNN($x_i$) represents the k nearest neighbor of sample $x_i$, KNN($x_i$,$k$) represents the $k$ nearest neighbor of sample $x_i$, and KNN similarity is as Eq. (7):

$$KNN\_Sim(x_i, x_j) = \left(\left|KNN(x_i) \cap KNN(x_j)\right|\right)/k \tag{7}$$

where $|KNN(x_i) \cap KNN(x_j)|$ means the intersection number of $k$ nearest neighbors of two samples. Clustering starts from each local high gravity center and uses KNN similarity as the measurement standard for extended clustering. When KNN similarity is higher than the similarity threshold $s$, the two samples will be divided into the same class. In this process, multiple sub clusters will be merged, because there will be more high gravity centers than the actual clusters. The flow of the whole algorithm is shown in the Table 1.

**Table 1.** The flow of IDPC-GP.

| IDPC-GP Algorithm |
|---|
| **Input**: $X$, $k$ , $s$; |
| **Output**: Array *cluster*; |
| 1. Establish $k$ nearest neighbor matrix KNN of sample set $X$; |
| 2. Calculate the local gravity value of each sample point by Eq.(6); |
| 3. The samples whose gravity value is higher than that of all k nearest neighbors are defined as local high-density centers, and the set of high-density centers $C=\{c_1, c_2,..., c_n\}$ is get, where $1 \le n \le N$. |
| 4. Calculate KNN similarity between each pair of samples with Eq.(7). |
| 5. Taking each high gravity center as the starting point and KNN similarity as the measurement standard, extended clustering is carried out. Tag array *cluster* records clustering. |
| 6. If high gravity centers are found to belong to the current cluster during clustering, they will be merged together. |
| 7. After all clustering based on high gravity center is completed, the remaining points are regarded as outliers. |

# 4 Experiments

## 4.1 Datasets

The test datasets utilized in the experiments are four representative graphic datasets from UCI: *Jain*, *spiral*, *unbalance*, *Flame*, *R15*, *aggregation*, *PathBased* and *4k2_far*. It contains banded, striped, unbalanced, multi type combination and other data sets, which can comprehensively test the performance of the approach.

## 4.2 Parameter Test

The IDPC-GP algorithm involves a core parameter $k$. The gravity center is determined by the distribution of *KNN*, that is, the parameters are related to the value of $k$. In order to verify the robustness of the IDPC-GP to the $k$, the influence of this parameter on the approach effect is specially tested. Based on the idea of IDPC-GP, as long as the high gravity center group covers all natural clusters, all cluster structures can be found and the clustering task can be completed. In the Table 2, we tested the number of high gravity centers mined under different $k$ values on 8 UCI data sets, and these local centers cover all natural clusters.

**Table 2.** Effective coverage test of high gravity center.

| $k$ | 5 | 6 | 7 | 8 | 9 | 10 | 15 | 20 | 25 |
|---|---|---|---|---|---|---|---|---|---|
| Jain | 36 | 31 | 29 | 25 | 21 | 18 | 14 | 12 | 10 |
| Spiral | 31 | 16 | 10 | 6 | 5 | 4 | 3 | 3 | 3 |
| Unbalance | 648 | 545 | 462 | 402 | 353 | 315 | 196 | 129 | 89 |
| Flame | 23 | 17 | 11 | 10 | 9 | 7 | 5 | 4 | 3 |
| R15 | 46 | 41 | 35 | 32 | 22 | 19 | 16 | 15 | 15 |
| Aggregation | 90 | 75 | 61 | 55 | 45 | 40 | 22 | 18 | 16 |
| PathBased | 35 | 30 | 22 | 18 | 14 | 12 | 4 | 4 | 2 |
| 4k2_far | 37 | 30 | 28 | 20 | 18 | 11 | 9 | 6 | 6 |

In Table 2, the first row shows different values of $k$ in [5, 25], and the other rows show the number of gravity centers on different data sets and at different values of $k$. All the high gravity centers in the above are natural clusters of full coverage data sets. The experimental results show that in a relatively wide range of $k$, the high gravity centers can effectively cover all natural clusters, which provides accurate positioning for the subsequent discovery of clusters.

## 4.3 Performance Comparison Test

From the above experimental results, we can see that the clustering effect of IDPC-GP algorithm in unbalanced, non-spherical and banded unbalanced data sets is satisfactory.

Of course, the DPC algorithm can achieve the same effect on these data sets, but it has high requirements for parameter combination. While the IDPC-GP algorithm is insensitive in parameter settings and has stronger parameter tolerance. To prove the ability of the IDPC-GP to discover clusters after discovering local centers, the clustering effects of several algorithms are tested and the results are displayed visually as shown in the Fig. 5. Obviously, in the face of different types of data sets, the clustering effect is satisfactory.

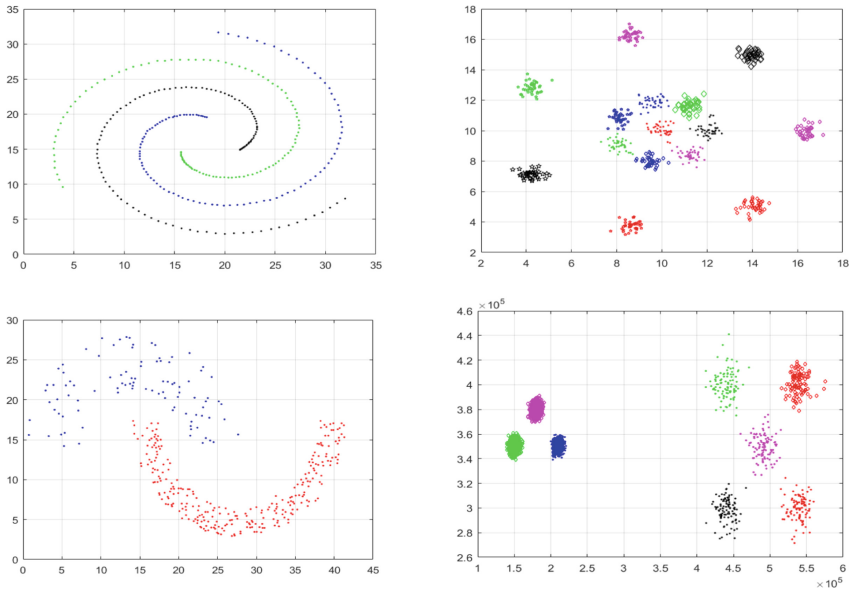

**Fig. 5.** Clustering effect of IDPC-GP algorithm on shape data sets.

As shown in Table 3, taking *Jain* data set as an example, with the change of $k$ value, the change of $s$ value in Fig. 5 can be achieved. Here, we show the case of making $s$ value as constant as possible.

**Table 3.** Collocation value of $k$ and $s$.

| Arguments | | | |
|---|---|---|---|
| $k$ | **5−11** | **12−22** | **23−25** |
| $s$ | 0.15 | 0.4 | 0.5 |

Finally, to test the robustness of the IDPC-GP to parameter s, we take data Jain as an example. When the results of clustering are optimal, the values range of $k$ and $s$ is shown in the Table 3. The $s$ value corresponding to each $k$ value is not unique. For example, when $k = 15$, the value range of the most $s$ value reached by the IDPC-GP algorithm on

the *Jain* dataset is [0.2, 0.45]. Compared with DPC algorithm, the parameter $d_c$ value can only be in [10, 15], and the parameter selection space is larger.

## 5 Conclusion

Aiming at the problem that DPC algorithm cannot automatically determine the number of high-density centers and is sensitive to parameters, a clustering algorithm IDPC-GP based on local data gravity center and KNN similarity extension is proposed. The local data gravity center can well cover the main clusters in the data space, and can accurately find the distribution of all clusters. Taking each gravity center as the starting point, extended clustering based on KNN similarity can complete the automatic clustering process without human intervention. The value of the important parameter $k$ involved in the algorithm has a very low impact on the effect of the algorithm, which improves the universality of the algorithm. Compared with DPC algorithm, it has some advantages in parameter adaptation and universality.

## References

1. Jian, H., Xu, E.: An improved density peak clustering algorithm. In: International Conference on Intelligent Data Engineering and Automated Learning, pp. 211–221 (2017)
2. Wang, Y., Wang, D., Zhou, Y., et al.: VDPC: variational density peak clustering algorithm. Inf. Sci. **621**, 627–651 (2021)
3. Rui, Z., Tao, D., Shouning, Q., et al.: Adaptive density-based clustering algorithm with shared KNN conflict game. Inf. Sci. **565**(5), 344–369 (2021)
4. Ren, C., Sun, L., Yu, Y., et al.: Effective density peaks clustering algorithm based on the layered k-nearest neighbors and subcluster merging. IEEE Access **99**, 1 (2020)
5. Tao, L.I., Hongwei, G.E., Shuzhi, S.U.: Density peaks clustering by automatic determination of cluster centers. J. Front. Comput. Sci. Technol. **10**(11), 1614–1622 (2010)
6. Jiang, P., Zeng, Q.: An Improved density peak clustering algorithm based on grid. Comput. App. Softw. (2019)
7. Jinyin, C., Xiang, L., Haibing, Z., et al.: A novel cluster center fast determination clustering algorithm. Appl. Soft Comput. **57**, 539–555 (2017)
8. Liu, R., Huang, W., Fei, Z., et al.: Constraint-based clustering by fast search and find of density peaks. Neurocomputing **330**(FEB.22), 223–237 (2019)
9. Rodriguez, A., Laio, A.: Machine learning. Clustering by fast search and find of density peaks. Science **344**(6191), 1492–1496 (2014)
10. Peng, L., Zhang, H., Chen, Y., Yang, B.: Imbalanced traffic identification using an imbalanced data gravitation-based classification model. Comput. Commun. **102**, 177–189 (2017)