



ELSEVIER

Contents lists available at ScienceDirect

Information Sciences

journal homepage: [www.elsevier.com/locate/ins](http://www.elsevier.com/locate/ins)

# Innovative multi-granularity granular-balls rough set for feature selection: Driving generalized multi-granularity rough set evolution with Zentropy integration

Weirui Ye , Weihua Xu\*

College of Artificial Intelligence, Southwest University, Chongqing, 400715, PR China

## ARTICLE INFO

## Keywords:

Feature selection  
 Granular-ball computing  
 Granular computing  
 Generalized multi-granularity rough set

## ABSTRACT

Granular-ball computing provides an efficient, robust, and scalable framework for granular computing tasks. The original granular-balls generation method, akin to  $k$ -means clustering, begins by aggregating the entire dataset into a single granular-ball and then iteratively subdividing it into smaller units. While convenient, this approach inherits the limitations of  $k$ -means, performing poorly on partially distributed datasets. Additionally, it generates granular-balls with a single granularity, limiting its capacity to fuse data across multiple granular space (subset of the attribute power set) and resulting in information loss. To address these challenges, we propose a novel granular-balls generation method based on neighbor search. Building on this, we define multi-granularity granular-balls, enabling the generation of granular-balls across multiple granular space to better capture diverse data distributions. Furthermore, we explore the practical applications of multi-granularity granular-balls by introducing a feature selection method called generalized multi-granularity granular-balls rough set (GMG-GBRS). This method integrates multi-granularity granular-balls with generalized multi-granularity rough set with Zentropy. By utilizing the sample-reduction capabilities and multiple granular space filtering during granular-balls generation, GMG-GBRS effectively reduces the domain size of rough set and minimizes the required multiple granular space, thereby significantly enhancing computational efficiency. Moreover, the superior data fusion capabilities of multi-granularity granular-balls, combined with their adaptive ability to generate neighborhood relationships, synergize with Zentropy's precise uncertainty measurement across diverse granularity levels in information systems, thereby enhancing the efficacy of feature selection.

## 1. Introduction

Granular computing is an emerging paradigm in information processing that addresses complex information entities known as “information granules”, which are derived through data abstraction and knowledge extraction processes. Although granular computing lacks a formal axiomatic definition, several widely recognized models have been established, including rough set theory [1,2], quotient space theory [3], and cloud model theory [4]. These models have enabled granular computing to find extensive applications in feature selection [5], information fusion [6], and anomaly detection [7]. These capabilities are particularly relevant to

\* Corresponding author.

E-mail addresses: [reanye233@gmail.com](mailto:reanye233@gmail.com) (W. Ye), [chxuwh@gmail.com](mailto:chxuwh@gmail.com) (W. Xu).

<https://doi.org/10.1016/j.ins.2025.122411>

Received 1 April 2025; Received in revised form 3 June 2025; Accepted 5 June 2025

Available online 10 June 2025

0020-0255/© 2025 Elsevier Inc. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

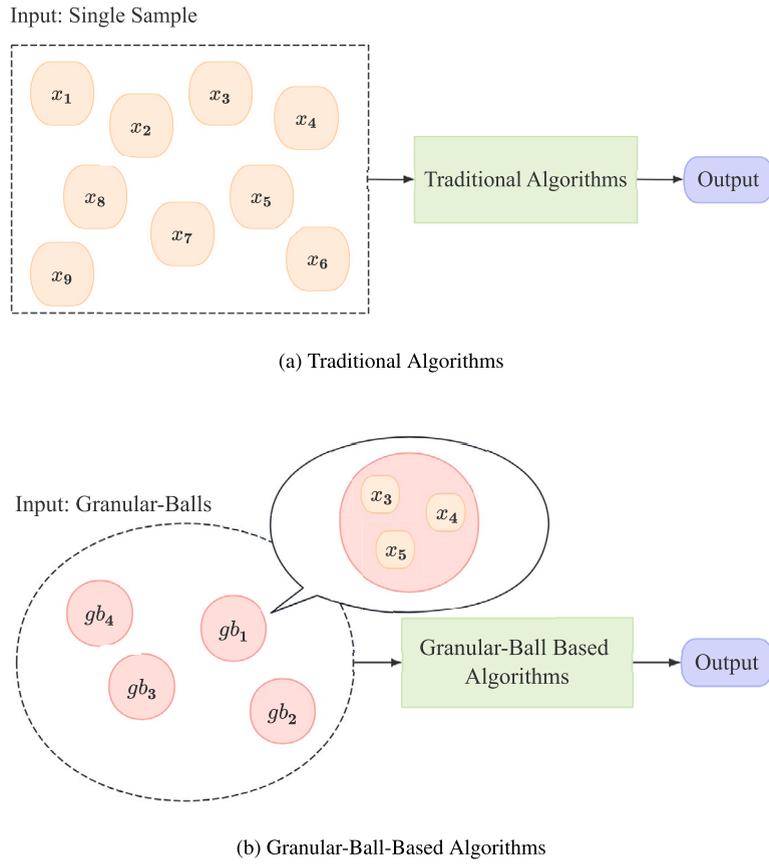


Fig. 1. Differences Between Traditional and Granular-Ball-Based Algorithms.

data-intensive domains such as bioinformatics [8,9], where managing uncertainty and extracting knowledge from high-dimensional and heterogeneous data are critical challenges.

A significant advancement in this field is the development of granular-ball computing, inspired by the discovery of “global priority features of human cognition” [10]. Based on this cognitive mechanism, Wang et al. proposed multi-granularity cognitive computing [11], which was further extended by Xia and Wang into granular-ball computing, demonstrating advantages in efficiency, robustness, and interpretability [12]. In granular-ball computing, multiple point data are aggregated into “balls”, which are then processed by granular-ball-based algorithms, as illustrated in Fig. 1. This strategy enhances both robustness and computational efficiency [13]. In addition to granular-ball methods, several other non-ball-based algorithms have also shown strong noise resistance, particularly in high-dimensional or noisy environments [14]. These approaches provide complementary perspectives on robust information processing, and together they broaden the toolbox for handling uncertainty in real-world data.

Granular-ball computing has since been widely applied in classification and clustering tasks, demonstrating its versatility and effectiveness. For instance, Xia et al. integrated granular-ball computing with support vector machines (SVM), resulting in granular-ball-based SVMs [15,16]. They further introduced controllable multi-granularity SVM (con-MGSVM) and support vector regression (con-MGSVR), significantly improving SVM performance [17]. In clustering, Xia et al. proposed ball  $k$ -means, a granular-ball-enhanced variant of the exact  $k$ -means algorithm, which accelerates convergence [18]. Building on this, Xie et al. developed a granular-ball-based spectral clustering method, which improves similarity matrix construction while reducing computational complexity [19].

However, it is important to clarify the notion of multi-granularity in different contexts. In Xia’s work [15–18], multi-granularity often refers to varying granular-ball sizes. In contrast, multi-granularity in this study refers to computing in a multi granular space (subset of the attribute power set). This discrepancy reflects the lack of unified terminology in granular computing. Rather than being contradictory, these definitions illustrate the contextual flexibility and adaptability of the granular computing paradigm.

Beyond classification and clustering, granular-ball computing has also contributed significantly to feature selection. Xia et al. introduced granular-ball neighborhood rough set (GBNRS), which adaptively form object-specific neighborhoods in  $O(N)$  time, offering more flexibility than traditional neighborhood rough set [20]. They subsequently proposed granular-ball rough set (GBRS), which combine the structural strengths of granular-ball computing with the descriptive capabilities of Pawlak and neighborhood rough set, facilitating the processing of continuous data and equivalence-based knowledge representation [21].

The scope of granular-ball computing further extends to multi-label learning. Qian et al. proposed a granular-ball and label distribution-based feature selection method, which clusters multi-label data into adaptive granular structures and uses label enhancement to convert logical labels into probabilistic distributions through instance-level similarity analysis [22]. To handle missing labels, Shu et al. designed a granular-ball-based mutual information feature selection algorithm, which improves classification accuracy via label recovery and granular-ball mutual information computation [23].

Furthermore, granular-ball computing has been effectively integrated with fuzzy rough set models. For label distribution learning, a granular-ball-based fuzzy rough set (GBFRS) method was proposed to address label ambiguity and dimensionality, achieving strong performance across twenty-two benchmark datasets [24]. In streaming data settings, a fuzzy neighborhood granular-ball rough set (FNGBRS) model was introduced by combining Canopy clustering and granular-ball computing, enabling parameter-free online group feature selection with improved efficiency and stable classification results [25]. Additionally, granular-ball computing enhances the robustness and scalability of fuzzy rough set models in high-dimensional spaces, outperforming traditional baselines [26].

Despite these advances, most rough set-based feature selection algorithms remain constrained to single-granularity frameworks [27,28]. To address this limitation and capture richer information from multiple granular space, multi-granularity rough set models have been proposed. For instance, Sun et al. developed a fuzzy neighborhood multi-granulation rough set-based feature selection framework that integrates algebraic and information-theoretic measures to improve classification stability [29]. Xu et al. presented a generalized multi-granulation neighborhood rough set model using matrix representations and introduced a new entropy measure for guiding heuristic feature subset selection [30]. Likewise, Zhang et al. proposed a feature selection algorithm based on generalized multi-granulation fuzzy neighborhood rough set (GMFNRS), which leverages fuzzy neighborhood entropy to improve uncertainty quantification and support efficient feature selection in complex environments [31].

Uncertainty measurement is a critical component in rough set-based feature selection algorithms, directly influencing their performance. To improve the accuracy of uncertainty assessment in information systems, researchers have introduced information entropy into rough set models with considerable success [32,33]. In this study, we adopt Zentropy [34,35] as our uncertainty measure. The term ‘‘Zentropy’’ originates from the German word ‘‘Zustandssumme’’, meaning the sum over different states or scales. This measure focuses on uncertainty across various granularity levels to design more robust feature evaluation functions. Compared with other entropy-based measures, Zentropy offers a more comprehensive and resilient characterization of uncertainty in decision information systems.

Despite the rapid progress in both multi-granularity rough set and granular-ball rough set, research on integrating granular-ball rough set into multi-granularity frameworks has largely stagnated. This stagnation stems primarily from the limitations of the original granular-balls generation method [12], which lacks the ability to construct granular-balls in multiple granular space. Moreover, its partitioning-based strategy performs poorly on datasets with complex or irregular distributions, thus limiting its applicability. Although recent studies have improved the efficiency of granular-ball generation [19,36], they fail to address the fundamental challenge of effectively representing data at multiple granularities. These unresolved issues motivate our work, which seeks to develop a more flexible and robust granular-ball generation method tailored for multi-granularity modeling.

Therefore, to optimize granular-ball generation and enable its integration into multi-granularity rough set for enhanced performance, we have made the following contributions:

1. We propose a novel granular-balls generation algorithm based on neighbor search. This method groups each data sample and its neighbors into a granular-ball, incrementally generating granular-balls. Unlike partitioning-based approaches, our method adapts to arbitrary data distributions, significantly improving data fusion accuracy.
2. We define the concept of multi-granularity granular-balls and devise a generation algorithm for them, building upon the neighbor search-based granular-balls generation approach. The algorithm enables multi-granularity granular-balls to represent data in multiple granular space while filtering out invalid granularities, expediting subsequent multi-granularity feature selection algorithms. The process is illustrated in Fig. 2.
3. We develop a feature selection algorithm based on generalized multi-granularity rough set, incorporating Zentropy [34,35] to accurately measure uncertainty across diverse granularity levels. Our approach reduces sample size and granularity complexity through granular-balls generation, enabling efficient granularity screening and neighborhood relationship establishment, thereby improving computational efficiency and feature selection effectiveness.

Compared with the traditional feature selection algorithm based on rough set, our proposed algorithm has the following advantages:

1. By generating multi-granularity granular-balls based on neighbor search, granular-balls can be generated in arbitrarily distributed data sets, forming a more accurate granular-balls neighborhood relationship, so that the granular-balls rough set can calculate the upper and lower approximations more accurately, and combined with Zentropy, it can more accurately measure the uncertainty of the information system, thereby improving the effect of feature selection.
2. Compared with the traditional multi-granularity rough set, which needs to calculate the entire multiple granular space (i.e., the power set of the attribute set), in the process of generating multi-granularity granular-balls, the granularity will be screened, thereby eliminating the granularity that cannot generate the granular-ball neighborhood relationship, thereby simplifying the size of the granular space, increasing the granularity that needs to be calculated for the multi-granular rough set, and improving the computational efficiency of the multi-granular rough set.

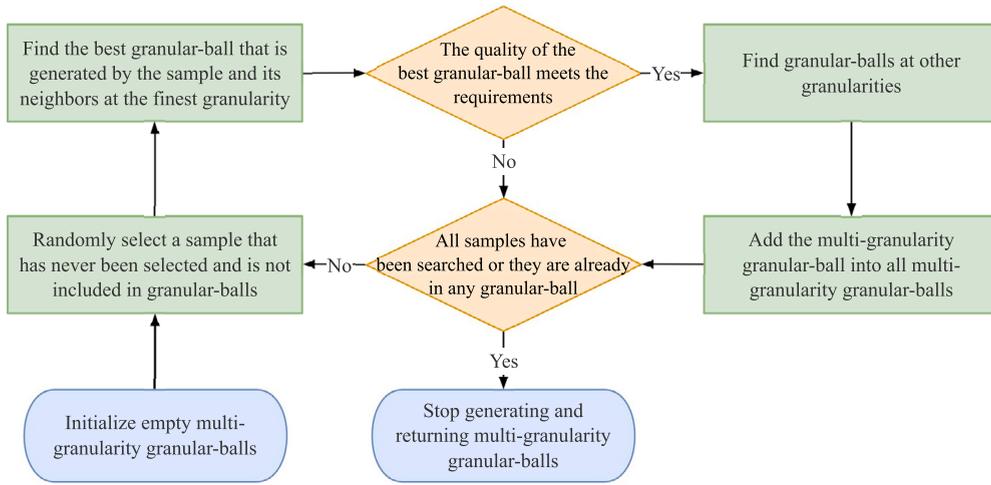


Fig. 2. Process for Generating Multi-Granularity Granular-Balls (Algorithm 2).

Table 1  
Summary of Notations.

Symbol	Description
$U$	Set of all samples, $U = \{x_1, x_2, \dots, x_m\}$ .
$D$	Set of decision attributes (labels) for all samples, $D = \{d_1, d_2, \dots, d_m\}$ .
$AT$	Complete set of condition attributes, $AT = \{a_1, \dots, a_n\}$ .
$A$	A subset of $AT$ .
$\mathcal{P}(A)$	Power set of $A$ , i.e., the set of all its subsets.
$AP$	Multi-granular space, $AP = \{A_1, A_2, \dots, A_q\} \subseteq \mathcal{P}(A)$ .
$\mathcal{GB}$	Set of multi-granularity granular-balls, $\mathcal{GB} = \{g^1, g^2, \dots, g^p\}$ .
$g^i$	A multi-granularity granular-ball composed of granular-balls across different granularities, $g^i = \{g^{i_1}, g^{i_2}, \dots, g^{i_p}\}$ .
$A_j$	A single granularity, where $A_j \in AP$ .
$g^{A_j}$	Granular-ball under granularity $A_j$ , defined as a quintuple $(\hat{U}_i^{A_j}, c_i^{A_j}, r_i^{A_j}, d_i^{A_j}, p_i^{A_j})$ .
$\hat{U}_i^{A_j}$	Set of samples contained within the granular-ball.
$c_i^{A_j}$	Center of the granular-ball.
$r_i^{A_j}$	Radius of the granular-ball.
$d_i^{A_j}$	Label of the granular-ball, determined by the majority class among contained samples.
$p_i^{A_j}$	Purity of the granular-ball, i.e., the proportion of samples sharing the majority label.
$Ra(g^{A_j})$	Overlap ratio of the granular-ball.
$P_{X_k}^{A_j}(x)$	Support characteristic function indicating whether the neighborhood of $x$ is contained in decision class $X_k$ .
$\Delta(x, y)$	Euclidean distance between samples $x$ and $y$ .
$\beta$	Support threshold, $\beta \in (0.5, 1]$ .
$GM_{AP}^\beta(X_k)$	Lower approximation of decision class $X_k$ in the generalized multi-granularity granular-ball rough set.
$GM_{AP}^\beta(X_k)$	Upper approximation of decision class $X_k$ in the generalized multi-granularity granular-ball rough set.
$Z(A, D)$	Zentropy with respect to attribute set $A$ and decision attribute set $D$ .
$IM(a, A, D)$	Inner significance measure of attribute $a \in A$ .
$SM(a, A, D)$	Outer significance measure of attribute $a \in A$ .

3. Similar to the original granular-balls rough set, multi-granularity granular-balls can simplify the domain of the rough set (from the number of samples  $|U|$  to the number of granular-balls  $|\mathcal{GB}|$ ), thereby improving the computational efficiency of the rough set.

The remainder of this paper is organized as follows. Section 2 introduces the basics of granular-ball computing and generalized multi-granularity rough set. Section 3 presents the definition and generation method for multi-granularity granular-balls based on neighbor search. Section 4 proposes generalized multi-granularity granular-balls rough set, integrating granular-balls with generalized multi-granularity rough set. In Section 5, we evaluate the efficiency of granular-balls generation and classification performance. Finally, Section 6 summarizes our findings and discusses future research directions.

Table 1 is a summary of the symbols involved in the algorithm proposed in this paper.

## 2. Preliminaries

In this section, we introduce the fundamental concepts of granular-ball computing and the generalized multi-granularity rough set.

### 2.1. Granular-ball computing

Most existing feature selection algorithms operate at the finest level, taking individual points or pixels as inputs. However, this point-level perspective diverges from the coarse-to-fine nature of human cognition, which tends to grasp global structures before focusing on fine details. To bridge this gap, Xia et al. proposed Granular-Ball Computing (GBC), a novel data analysis framework rooted in the principles of granular computing [12]. Rather than representing data as isolated discrete points, GBC introduces the concept of granular-balls—hyperspherical regions in the feature space that encapsulate local data structures. This coarse-grained representation captures not only the central tendency of data but also preserves essential interrelationships among similar samples. An illustrative comparison between traditional methods and granular-ball-based algorithm is shown in Fig. 1.

To rigorously define a granular-ball, consider a labeled dataset  $U = \{x_1, x_2, \dots, x_m\}$ , where  $d_i$  denotes the label associated with sample  $x_i$ . A granular-ball  $gb$  is represented as a quintuple:

$$gb = (\hat{U}, c, r, \hat{d}, p), \tag{1}$$

with the components defined as follows:

- $\hat{U} \subseteq U$  is the set of data samples contained in the granular-ball.
- $c$  is center of the granular-ball, computed as:

$$c = \frac{1}{|\hat{U}|} \sum_{x_i \in \hat{U}} x_i, \tag{2}$$

- $r$  is radius of the granular-ball, which is the average distance from the samples to the center:

$$r = \frac{1}{|\hat{U}|} \sum_{x_i \in \hat{U}} \|x_i - c\|, \tag{3}$$

where  $\|\cdot\|$  denotes the L2 norm.

- $\hat{d}$  is label of the granular-ball, which is the most frequent label among the samples in the ball,

$$\hat{d} = \text{Mo}(\{d_i \mid x_i \in \hat{U}\}), \tag{4}$$

where  $\text{Mo}(\cdot)$  denotes the mode function.

- $p$  is called purity, which is the ratio of samples in the ball that match the majority label:

$$p = \frac{|\{d_i \mid d_i = \hat{d}\}|}{|\hat{U}|}. \tag{5}$$

This representation enables GBC to perform data modeling at multiple levels of abstraction. By grouping similar samples into compact and semantically meaningful units, it significantly reduces computational complexity, making it well-suited for large-scale datasets. Moreover, in the context of rough set-based feature selection, the use of granular-balls supports more adaptive and meaningful neighborhood identification, thereby enhancing approximation quality and algorithmic efficiency.

To formalize the neighborhood relation induced by granular-balls, let  $GB = \{gb_1, gb_2, \dots, gb_p\}$  be the set of granular-balls generated from  $U$ . The granular-ball neighborhood relation  $\mathcal{R}$  is defined as the binary relation:

$$\mathcal{R} = \{(gb_i, x_j) \in GB \times U \mid x_j \in \hat{U}_i\}. \tag{6}$$

In other words, a sample  $x_j$  is related to a granular-ball  $gb_i$  if it belongs to  $gb_i$ .

From this definition, it follows that the neighborhood of any granular-ball  $gb_i \in GB$  consists precisely of the samples it encloses:

$$\mathcal{R}(gb_i) = \hat{U}_i. \tag{7}$$

The generation of granular-balls is conducted in a top-down hierarchical fashion. Initially, the entire dataset is treated as a single coarse ball. This ball is then recursively split into smaller granular-balls based on sample distribution and label purity, until all resulting granular-balls satisfy a predefined purity threshold. This adaptive, data-driven strategy not only improves representational fidelity but also provides a natural pathway for multi-level reasoning. The overall process of granular-ball generation is illustrated in the flowchart of the original GBC algorithm in Fig. 3.

### 2.2. Generalized multi-granularity rough set

The classical rough set theory, proposed by Pawlak in the 1980s [1], provides a powerful mathematical framework for dealing with imprecise, uncertain, and incomplete information. However, it fundamentally relies on equivalence relations (or indiscernibility relations), which makes it inherently more suitable for categorical or discrete data. In real-world scenarios involving continuous attributes and numerical data, this assumption often becomes limiting.

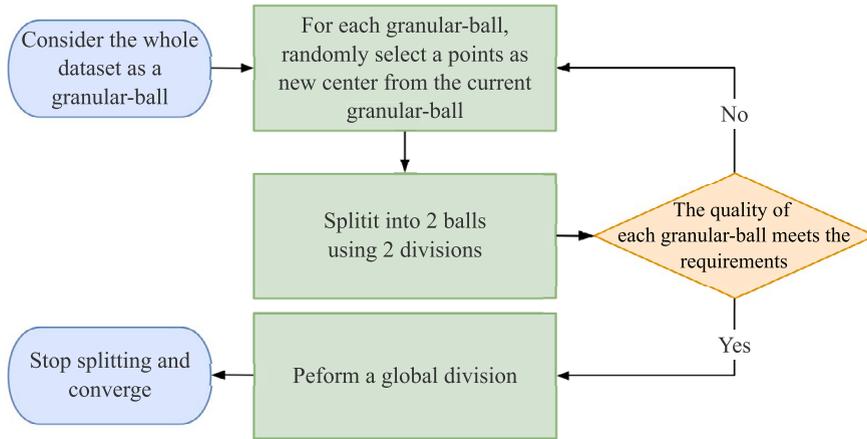


Fig. 3. Process for Generating Single-Granularity Granular-Balls (Origin).

To overcome this constraint and better handle real-valued data, the Neighborhood Rough Set (NRS) [2] model was introduced. Rather than relying on equivalence classes, the NRS model defines object similarity through distance-based neighborhoods. Specifically, given a distance function  $\Delta : U \times U \rightarrow \mathbb{R}^+$ , the  $\delta$ -neighborhood of a sample  $x \in U$  is defined as:

$$[x]_\delta = \{y \in U \mid \Delta(x, y) \leq \delta\}, \tag{8}$$

where  $\delta > 0$  is a user-defined neighborhood radius that controls the granularity of approximation. This mechanism allows the NRS model to adapt to noisy environments and gradual boundary regions commonly seen in numerical datasets.

Given a partition  $U/D = \{X_1, X_2, \dots, X_s\}$  induced by decision attribute, the lower and upper approximations under the NRS framework are defined as:

$$\underline{R}_\delta(X_k) = \{x \in U \mid [x]_\delta \subseteq X_k\}, \tag{9}$$

$$\overline{R}_\delta(X_k) = \{x \in U \mid [x]_\delta \cap X_k \neq \emptyset\}. \tag{10}$$

By tuning the neighborhood radius  $\delta$ , the NRS model offers a flexible balance between approximation precision and tolerance to data uncertainty, making it widely applicable in tasks such as feature selection, pattern recognition, and data mining.

Despite its advantages, the single-granularity nature of traditional NRS may fail to capture multi-scale structural information embedded in complex datasets. To address this issue, researchers have proposed the multi-granulation rough set (MGRS) models [29–31], which incorporate multiple granular perspectives to enhance robustness. In particular, the traditional MGRS approaches—namely the optimistic and pessimistic models—are based on whether an object satisfies approximation conditions across all or at least one granularity. However, the pessimistic model is often too strict, while the optimistic model is overly permissive, which can both lead to poor performance in practical applications.

To strike a better balance between these two extremes, the Generalized Multi-Granulation Rough Set (GMGRS) model was introduced. It allows partial agreement across multiple granularities through a support threshold, thus offering a more flexible and adaptive approximation framework.

It is worth noting, however, that the notion of multi-granularity can vary significantly across different research contexts. For example, in Xia’s work [15–18], multi-granularity typically refers to variations in granular-ball sizes. In contrast, this study adopts a different perspective, where multi-granularity is defined in terms of computations carried out over a multi-granular space—that is, a collection of attribute subsets derived from the power set of the full attribute set.

Formally, let  $A$  be an attribute set and  $\mathcal{P}(A)$  its power set. Let  $AP = \{A_1, A_2, \dots, A_q\} \subseteq \mathcal{P}(A)$  denote a multi-granular space, where each  $A_j \in AP$  represents a single granularity. The neighborhood class of a sample  $x$  under  $A_j$  is defined as:

$$[x]_\delta^{A_j} = \{y \in U \mid \Delta(x, y) \leq \delta\}. \tag{11}$$

Given a partition  $U/D = \{X_1, X_2, \dots, X_s\}$  induced by decision attribute, define the support characteristic function  $P_{X_k}^{A_j}(x)$  as:

$$P_{X_k}^{A_j}(x) = \begin{cases} 1, & \text{if } [x]_\delta^{A_j} \subseteq X_k, \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

Based on this, for a support threshold  $\beta \in (0.5, 1]$ , the generalized multi-granulation lower and upper approximations are defined as:

$$\overline{GR_{AP}^\beta}(X_k) = \left\{ x \in U \mid \frac{1}{|AP|} \sum_{A_j \in AP} P_{X_k}^{A_j}(x) \geq \beta \right\}, \tag{13}$$

$$\overline{GR_{AP}^\beta}(X_k) = \left\{ x \in U \mid \frac{1}{|AP|} \sum_{A_j \in AP} \left( 1 - P_{X_k^c}^{A_j}(x) \right) > 1 - \beta \right\}, \tag{14}$$

in which  $X_k^c$  is the complement of  $X_k$ .

This generalized model enhances classical and neighborhood-based rough set by incorporating multiple views and tolerating partial consistency, making it especially effective in high-dimensional, noisy, or heterogeneous data environments.

### 3. Generation of multi-granularity granular-balls

Traditional granular-balls are limited to single-granularity data fusion, lacking the ability to integrate multi-granularity information. To address this limitation and enhance their capability to capture multi-granularity characteristics, we extend the concept of single-granular granular-balls to multi-granularity granular-balls, enabling effective information fusion across multiple granularities.

**Definition 1.** Given an information system  $IS = (U, AT \cup D, V, f)$ , where  $U = \{x_1, x_2, \dots, x_m\}$  and  $AT = \{a_1, a_2, \dots, a_n\}$ , let  $A \subseteq AT$ . The power set of  $A$  is denoted as  $\mathcal{P}(A) = \{A_1, A_2, \dots, A_{2^{|A|}}\}$ . Define a multiple granular space  $AP = \{A_1, A_2, \dots, A_q\} \subseteq \mathcal{P}(A)$ . For any  $A_j \in AP$ , the set of multi-granularity granular-balls is defined as:

$$\mathcal{GB} = \{g\delta_1, g\delta_2, \dots, g\delta_p\}, \tag{15}$$

$$g\delta_i = \{gb_i^{A_1}, gb_i^{A_2}, \dots, gb_i^{A_q}\}, \quad i \leq p. \tag{16}$$

Here,  $gb_i^{A_j} = (\hat{U}_i^{A_j}, c_i^{A_j}, r_i^{A_j}, \hat{d}_i^{A_j}, p_i^{A_j}) \in g\delta_i$  represents a single-granularity granular-ball as introduced in Subsection 2.1, where  $A_j$  denotes the granularity. Unlike the earlier definition in Subsection 2.1, the radius  $r_i^{A_j}$  is computed as:

$$r_i^{A_j} = \frac{\|x_a^{A_j} - c_i^{A_j}\| + \|x_b^{A_j} - c_i^{A_j}\|}{2}. \tag{17}$$

Here,  $x_a$  and  $x_b$  represent the  $(|\hat{U}|)$ -th and  $(|\hat{U}| + 1)$ -th nearest neighbors of  $c_i^{A_j}$ , respectively, and  $\|\cdot\|$  denotes the L2 norm.

To facilitate computation, for any multi-granularity granular-ball  $g\delta_i = \{gb_i^{A_1}, gb_i^{A_2}, \dots, gb_i^{A_q}\}$ , we define the set of granularities  $\Omega$  as:

$$\Omega(g\delta_i) = \{A_1, A_2, \dots, A_q\}, \quad q \leq 2^{|A|}. \tag{18}$$

We propose a novel granular-balls generation algorithm based on neighbor search, capable of producing both single-granularity and multi-granularity granular-balls. Typically, the neighbors of a sample exhibit homogeneity, sharing the same label and belonging to the same granular-ball. Based on this assumption, the  $k$ -nearest neighbors  $\hat{U}_i^{A_j}$  of a sample  $x_i^{A_j}$  are grouped into the same granular-ball  $gb_i^{A_j}$ . From these neighbors, we compute the center  $c_i^{A_j}$  and radius  $r_i^{A_j}$  to obtain a single-granularity granular-ball  $gb_i^{A_j} = (\hat{U}_i^{A_j}, c_i^{A_j}, r_i^{A_j}, \hat{d}_i^{A_j}, p_i^{A_j})$ .

When generating single-granularity granular-balls at the finest level, we maximize their size by choosing a sufficiently large  $k$ . However, excessively large  $k$  values may reduce granular-ball purity. To balance size and quality, we search for an optimal  $k$  value using binary search, ensuring the granular-ball satisfies a predefined purity threshold equals to 1 while maximizing its size.

To handle overlap among granular-balls, we introduce the overlap ratio to quantify the degree of overlap within the same granularity:

**Definition 2.** For a granular-ball set  $\mathcal{GB} = \{g\delta_1, g\delta_2, \dots, g\delta_p\}$ ,  $g\delta_i = \{gb_i^{A_1}, gb_i^{A_2}, \dots, gb_i^{A_q}\}$ ,  $i \leq p$ , and any granular-ball  $gb_i^{A_j} = (\hat{U}_i^{A_j}, c_i^{A_j}, r_i^{A_j}, \hat{d}_i^{A_j}, p_i^{A_j}) \in g\delta_i$ , the overlap ratio is defined as:

$$Ra(gb_i^{A_j}) = \frac{\left| \bigcup_{t \neq i}^p (\hat{U}_t^{A_j} \cap \hat{U}_i^{A_j}) \right|}{|\hat{U}_i^{A_j}|}. \tag{19}$$

After constructing the finest granular-balls, we identify additional granularities that satisfy the purity condition, enabling the creation of multi-granularity granular-balls  $\mathcal{GB}$ . To manage high-dimensional data efficiently, we limit the size of the multiple granular space to  $[\underline{N}, \overline{N}]$ , selecting granularities iteratively to ensure computational feasibility.

**Algorithm 1:** Generating a Multi-Granularity Granular-Ball.

---

**Input:** Size limitation of granular-ball  $[\underline{M}, \overline{M}]$ , sample  $x$ , multiple granular space  $AP$  and its size limits  $[\underline{N}, \overline{N}]$ .  
**Output:** Multi-granularity granular-ball and multiple granular space.

```

1  $left = \underline{M}$ ,  $right = \overline{M}$ ,  $mid = \lfloor \frac{left+right}{2} \rfloor$ ,  $gb_i^A = \emptyset$ ,  $g\mathcal{B}_i = \emptyset$ ;
2 while  $left < right$  do
3   Query  $mid$ -nearest neighbors  $\hat{U}_i^{tA}$  of  $x$ ;
4   Create a granular-ball  $gb_i^A = (\hat{U}_i^{tA}, c_i^A, r_i^A, \hat{d}_i^A, p_i^A)$  via Definition 1;
5   if  $p_i^A \geq 1$  then
6      $left = mid + 1$ ,  $mid = \lfloor \frac{left+right}{2} \rfloor$ ;
7     if  $gb_i^A = \emptyset$  or  $|\hat{U}_i^{tA}| > |\hat{U}_i^A|$  then
8        $gb_i^A = gb_i^A$ ;
9     end
10  else
11     $right = mid - 1$ ,  $mid = \lfloor \frac{left+right}{2} \rfloor$ ;
12  end
13 end
14 if  $gb_i^A = \emptyset$  or  $Ra(gb_i^A) > 0$  then
15   return None;
16 end
17  $g\mathcal{B}_i \leftarrow gb_i^A$ ;
18 for  $A_j \in AP$  do
19   Query  $mid$ -nearest neighbors  $\hat{U}_i^{tA_j}$  of  $c_i^{A_j}$ ;
20   Create a granular-ball  $gb_i^{A_j} = (\hat{U}_i^{tA_j}, c_i^{A_j}, r_i^{A_j}, \hat{d}_i^{A_j}, p_i^{A_j})$  via Definition 1;
21   if  $\hat{d}_i^{A_j} = \hat{d}_i^A$  and  $Ra(gb_i^{A_j}) \leq 0$  then
22      $g\mathcal{B}_i \leftarrow gb_i^{A_j}$ ;
23   end
24 end
25 if  $|AP \cap \Omega(g\mathcal{B}_i)| < \underline{N}$  then
26   return None;
27 else
28    $AP = AP \cap \Omega(g\mathcal{B}_i)$ ;
29   return  $g\mathcal{B}_i, AP$ ;
30 end

```

---

**Algorithm 2:** Multi-Granularity Granular-Balls Generation Based on Neighbor Search.

---

**Input:** Samples set  $U$ , attribute set  $A$ , target purity  $T$ , size limitation of any granular-ball  $[\underline{M}, \overline{M}]$  and size limits of multiple granular space  $[\underline{N}, \overline{N}]$ .  
**Output:** Multi-granularity Granular-balls  $\mathcal{G}\mathcal{B}$ .

```

1  $S = \emptyset$ ,  $\mathcal{G}\mathcal{B} = \emptyset$ ;
2 Randomly select  $\overline{N}$  granularities as multiple granular space  $AP \subseteq \mathcal{P}(A)$ ;
3 while  $|S| < |U|$  do
4   Randomly select a single sample  $x \in (U - S)$ ;
5   Get a multi-granularity granular-ball  $g\mathcal{B}_i$  and multiple granular space  $AP$  via Algorithm 1;
6    $S \leftarrow x$ ;
7   Get  $gb_i^A = (\hat{U}_i^A, c_i^A, r_i^A, \hat{d}_i^A, p_i^A) \in g\mathcal{B}_i$ ;
8   if  $g\mathcal{B}_i$  is not None then
9      $S = S \cup \hat{U}_i^A$ ,  $\mathcal{G}\mathcal{B} \leftarrow g\mathcal{B}_i$ ;
10  end
11 end
12 return  $\mathcal{G}\mathcal{B}$ ;

```

---

The detailed algorithms for generating multi-granularity granular-balls are presented in Algorithm 1 and Algorithm 2.

Algorithm 1 primarily consists of a binary search loop and a subsequent loop through attribute sets. The binary search operates between  $\underline{M}$  and  $\overline{M}$ , resulting in a time complexity of  $O(\log(\overline{M} - \underline{M}))$  iterations. Within each iteration, the dominant operation is the k-nearest neighbors query, which we assume has a time complexity of  $f$ . The subsequent for-loop iterates over  $AP$ , which contains at most  $\overline{N}$  elements, and performs another k-nearest neighbors query in each iteration. Therefore, the overall time complexity of this algorithm is  $O(\log(\overline{M} - \underline{M}) \cdot f + |AP| \cdot f)$ .

Algorithm 2 has an outer while-loop that continues until all samples in  $U$  are processed. In the worst case, this could be  $O(|U|)$  iterations if each iteration only processes one sample. Each iteration calls the first algorithm and performs some set operations. The dominant term comes from the calls to the first algorithm, making the overall time complexity  $O(|U| \cdot (\log(\overline{M} - \underline{M}) \cdot f + |AP| \cdot f))$ .

#### 4. GMG-GBRS: generalized multi-granularity granular-balls rough set

The rough set theory is a crucial method for feature selection and represents a significant application within granular computing. However, prior research about granular-ball rough set has predominantly focused on single-granularity granular-balls, overlooking

the potential of multiple granular space. This limitation hinders the integration of granular-ball computing with multi-granularity rough set. To address this gap, and building on the multi-granularity granular-balls generation method introduced in the previous section, we propose the generalized multi-granularity granular-balls rough set (GMG-GBRS), which combines the strengths of multi-granularity granular-balls and generalized multi-granularity rough set.

**Definition 3.** Consider an information system  $IS = (U, AT \cup D, V, f)$  where  $U = \{x_1, x_2, \dots, x_m\}$  and  $AT = \{a_1, a_2, \dots, a_n\}$ . For any  $A \subseteq AT$ , the power set of  $A$  is denoted as  $\mathcal{P}(A) = \{A_1, A_2, \dots, A_{2^{|A|}}\}$ , and the multiple granular space is represented by  $AP \subseteq \mathcal{P}(A)$ .

The support characteristic function of the generalized multi-granularity granular-balls rough set (GMG-GBRS) is defined as:

$$P_{X_k}^{A_j}(gb_i^{A_j}) = \begin{cases} 1, & \text{if } \mathcal{R}(gb_i^{A_j}) \subseteq X_k, \\ 0, & \text{otherwise,} \end{cases} \tag{20}$$

in which  $\mathcal{R}(gb_i^{A_j}) = \mathring{U}_i^{A_j}$  is granular-ball neighbors of  $gb_i^{A_j}$ .

The generalized multi-granularity granular-ball lower and upper approximations are then expressed as follows:

$$\underline{GM}_{AP}^\beta(X_k) = \left\{ x \in U \mid \frac{1}{|AP|} \sum_{A_j \in AP} P_{X_k}^{A_j}(gb_i^{A_j}) \geq \beta \right\}, \tag{21}$$

$$\overline{GM}_{AP}^\beta(X_k) = \left\{ x \in U \mid \frac{1}{|AP|} \sum_{A_j \in AP} \left( 1 - P_{X_k^c}^{A_j}(gb_i^{A_j}) \right) > 1 - \beta \right\}. \tag{22}$$

In feature selection algorithms based on rough set, accurately measuring the uncertainty of the information system is essential for effective feature selection. To address this, we incorporate Zentropy [34,35] into GMG-GBRS. Zentropy introduces a novel perspective on uncertainty measurement grounded in entropy theory, emphasizing uncertainty across multiple granular levels. This approach shifts the focus from analyzing a single granular level to providing a more comprehensive evaluation, enabling a deeper understanding of uncertainty.

**Definition 4.** Given an information system  $IS = (U, AT \cup D, V, f)$ , let  $A \subseteq AT$ . Multi-granularity granular-balls, denoted as  $\mathcal{G}\mathcal{B} = \{gb_1, gb_2, \dots, gb_p\}$ , are generated using Algorithm 2. Here,  $gb_i = \{gb_i^{A_1}, gb_i^{A_2}, \dots, gb_i^{A_q}\}$ . For a partition  $U/D = \{X_1, X_2, \dots, X_s\}$ , let  $AP = \bigcap_{i=1}^p \Omega(gb_i)$ . The Zentropy is computed as follows:

$$Z(A, D) = - \sum_{k=1}^s p_k \log_2 p_k + \sum_{k=1}^s p_k Z_k, \tag{23}$$

where  $p_k = \frac{|X_k|}{|U|}$  is the probability of the  $k$ -th class at the decision level, and  $Z_k$  represents the entropy of the  $k$ -th class, which can be further decomposed at finer granular levels.

The entropy  $Z_k$ , reflecting the uncertainty at the approximation level, is represented as:

$$Z_k = - \sum_{l=1}^2 p_{kl} \log_2 p_{kl} + \sum_{l=1}^2 p_{kl} Z_{kl}, \tag{24}$$

where  $p_{k1} = \frac{|GM_{AP}^\beta(X_k)|}{|X_k|}$  and  $p_{k2} = \frac{|X_k - GM_{AP}^\beta(X_k)|}{|X_k|}$  represent the distribution of certainty and non-certainty sets in the class  $D_k$ .

The certainty in different granularities is defined as:

$$Z_{kl} = - \sum_{j=1}^{|AP|} p_{klj} \log_2 p_{klj} + \sum_{j=1}^{|AP|} p_{klj} Z_{klj}, \tag{25}$$

where  $p_{klj} = \frac{|\mathring{U}_i^{A_j}|}{|X_k|}$  represents the probability in the  $j$ -th granularity.

The certainty of an object depends on the relationship between its granular-ball neighborhood class and the target concept. Therefore,  $Z_{k1j}$  and  $Z_{k2j}$  can be further defined using granular-ball neighborhood classes as:

$$Z_{klj} = - \sum_{i=1}^{|N_{klj}|} p_{klji} \log_2 p_{klji} + \sum_{i=1}^{|N_{klj}|} p_{klji} Z_{klji}, \tag{26}$$

where  $N_{klj} = \frac{GM_{AP}^\beta(X_k)}{GM_{AP}^\beta(X_k)}$ ,  $N_{k2j} = X_k - \frac{GM_{AP}^\beta(X_k)}{GM_{AP}^\beta(X_k)}$ , and  $p_{klji} = \frac{|\hat{U}_i^{Aj}|}{\sum_{i=1}^p |\hat{U}_i^{Aj}|}$  represents the probability of the  $i$ -th granular-ball neighborhood class  $\hat{U}_i^{Aj}$  among all granular-ball neighborhood classes in  $N_{klj}$ .

Finally, the uncertainty at a finer object-specific level is defined as:

$$Z_{klji} = - \sum_{o=1}^2 p_{kljio} \log_2 p_{kljio}, \tag{27}$$

where  $p_{klji1} = \frac{|\hat{U}_i^{Aj} \cap X_k|}{|\hat{U}_i^{Aj}|}$  and  $p_{klji2} = \frac{|\hat{U}_i^{Aj} \cap X_k^c|}{|\hat{U}_i^{Aj}|}$  reflect the distribution of objects with decision labels in the granular-ball neighborhood classes.

By using Zentropy, we can more precisely characterize the uncertainty within the information system. Next, by calculating both the inner and outer importance measures, as defined in Definition 5, we can assess the impact of each individual attribute on the system’s uncertainty. Furthermore, by combining these measures with a heuristic search algorithm, we can identify a near-optimal feature combination that minimizes the uncertainty of the information system. The details of this feature selection algorithm are presented in Algorithm 3.

**Definition 5.** For a given set of samples  $U = \{x_1, x_2, \dots, x_m\}$ , consider an attribute subset  $a \in A \subseteq AT$ , and let  $U/D = \{X_1, X_2, \dots, X_s\}$ . The inner and outer significance measures of attribute  $a \in A$  are defined as follows:

$$IM(a, A, D) = Z(A - \{a\}, D) - Z(A, D), \tag{28}$$

$$SM(a, A, D) = Z(A, D) - Z(A \cup \{a\}, D). \tag{29}$$

---

**Algorithm 3:** Generalized Multi-Granularity Granular-Balls Rough Set (GMG-GBRS).

---

**Input:** An Information System  $IS = (U, AT \cup D, V, f)$ .  
**Output:** Selected features  $A$ .

```

1 Initialize  $A = \emptyset$ ;
2 for  $a \in AT$  do
3   Compute  $im = IM(a, AT, D)$  via Definition 5;
4   if  $im > 0$  then
5      $A \leftarrow a$ ;
6   end
7 end
8 while  $Z(A, D) > Z(AT, D)$  do
9   for  $a \in (AT - A)$  do
10    Compute  $sm = SM(a, A, D)$  via Definition 5;
11   end
12    $a_{max} = \arg \max_{a \in AT - A} (sm)$ ;
13    $A \leftarrow a_{max}$ ;
14 end
15 for  $a \in A$  do
16   if  $Z(A - \{a\}, D) < Z(A, D)$  then
17      $A = A - \{a\}$ ;
18   end
19 end
20 return  $A$ ;
```

---

Algorithm 3 features several loops over the attribute set  $AT$ . The initial for-loop computes importance measures for all attributes, resulting in  $O(|AT|)$  iterations. The while-loop continues until a condition on  $Z$  is met, with the number of iterations being data-dependent but bounded by  $O(|AT|)$  in the worst case. Each iteration contains a for-loop over the remaining attributes, leading to a potential  $O(|AT|^2)$  operations. The final for-loop performs another pass over the selected attributes, contributing  $O(|A|)$  operations. The overall time complexity is therefore  $O((|AT|^2 + |AT| + |A|) \cdot |U| \cdot (\log(\overline{M} - \underline{M}) \cdot f + |AP| \cdot f))$ .

**5. Experimental results and analysis**

In this section, we evaluate the efficiency and feature selection performance of our proposed algorithm through a series of experiments.

In Subsection 5.1, we visualize the granular-balls generation process of our proposed algorithm, highlighting its differences and advantages compared to the original granular-balls generation algorithm.

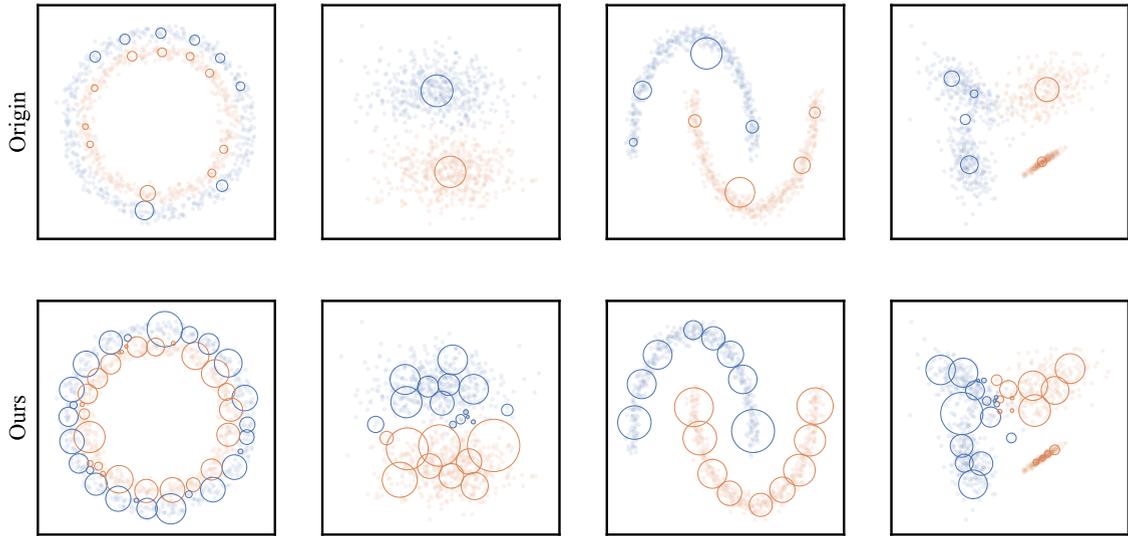


Fig. 4. Comparison of the Original Granular-Balls Generation and Our Method.

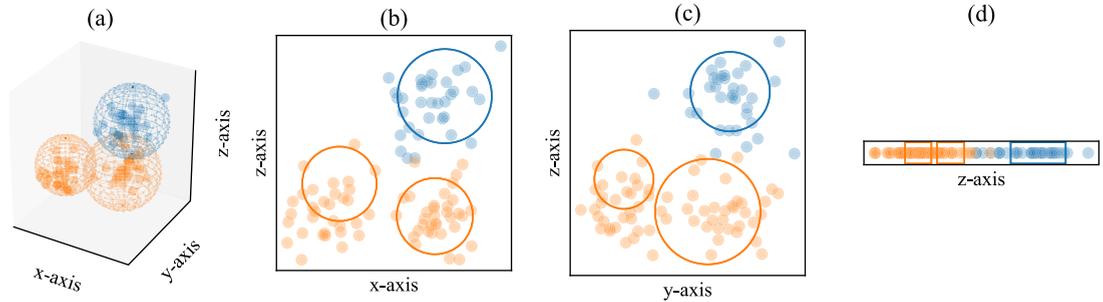


Fig. 5. Visualization of Multi-Granularity Granular-Balls.

In Subsection 5.2, we validate the effectiveness of the proposed feature selection algorithm by testing it alongside six existing feature selection methods on 18 datasets.

In Subsection 5.3, we assess the computational efficiency of our proposed generalized multi-granularity granular-balls rough set algorithm, comparing it with other feature selection algorithms. Additionally, we analyze the factors contributing to the high efficiency of our proposed approach.

All algorithms described in this paper are implemented in Python within the Anaconda Navigator environment. The experiments are conducted on a system equipped with an AMD Ryzen 7 5700G CPU operating at 3.80 GHz, 64.0 GB of memory, and running a 64-bit Windows 11 operating system.

### 5.1. Visualization of multi-granularity granular-balls generation

Granular-balls can represent data at various scales, but the original granular-balls generation algorithm, which is based on partitioning (similar to the  $k$ -means clustering algorithm), cannot effectively represent partially distributed datasets. To address this limitation, we propose a new granular-balls generation method based on neighbor search (Algorithm 2), which incorporates density-based clustering methods such as DBSCAN. This new method is more applicable to distributed datasets compared to the original granular-balls generation algorithm. The comparison between the original granular-balls generation method and our proposed neighbor search-based granular-balls generation method on several datasets is shown in Fig. 4.

From Fig. 4, we observe that, at the finest granularity level, our proposed granular-balls generation algorithm exhibits a stronger ability to represent data compared to the original method.

Furthermore, we introduce the concept of multi-granularity into granular-balls, allowing our granular-balls generation algorithm to generate granular-balls in multiple granular space, as shown in Fig. 5. This is a key distinction between the multi-granularity granular-balls we propose and the original single-granularity granular-balls, as our method can generate granular-balls across multiple granular space. This extension enables the use of our method in multi-granularity rough set, which is not possible with the original single-granularity granular-balls.

**Table 2**  
Details of Datasets.

No.	Dataset	Samples	Features	Classes	No.	Dataset	Samples	Features	Classes
1	Cancer	116	9	2	10	Semeion	1593	256	10
2	Darwin	174	450	2	11	Tunadromd	4464	241	2
3	Wine	178	13	3	12	Spambase	4601	57	2
4	Sonar	208	60	2	13	Quality	4898	11	7
5	Heart1	294	13	2	14	Mushroom	7535	22	2
6	Ionosphere	351	34	2	15	Htru3	8011	8	2
7	Urban	675	147	9	16	Mushroom1	8124	22	2
8	Qsar	1055	41	2	17	Bean	13611	16	7
9	Svmguide3	1284	22	2	18	Telescope	19020	10	2

Meanwhile, in the original multi-granularity rough set, the multiple granular space considered encompasses the power set of the attribute set. For instance, given the attribute set  $\{x, y, z\}$ , the power set would be  $\{\{x, y, z\}, \{x, y\}, \{x, z\}, \{y, z\}, \{x\}, \{y\}, \{z\}, \emptyset\}$ . However, as illustrated in Fig. 5, our multi-granularity granular-balls generation algorithm generates granular-balls only within a subset of this power set, specifically  $\{\{x, y, z\}, \{x, z\}, \{y, z\}, \{z\}\}$ . This is because our algorithm screens the granularity during the generation process, eliminating invalid granularities. Consequently, the subsequent multi-granularity feature selection algorithm no longer needs to consider the entire multiple granular space, resulting in a significant improvement in computational efficiency.

## 5.2. Feature selection evaluation of GMG-GBRS

In this section, we evaluate the effectiveness of the feature selection performance of our proposed GMG-GBRS algorithm. We selected 18 datasets for feature selection, which are described in Table 2, and they can be accessed via GitHub and UCI repositories.

During the data pre-processing stage, we applied min-max normalization to standardize all data and replaced any missing values with 0. The normalization formula is given as:

$$a(x_i) = \frac{a(x_i) - \min a(x)}{\max a(x) - \min a(x)}, \quad (30)$$

where  $a(x_i)$  represents the value of attribute  $a$  for sample  $x_i$ .

This normalization technique is effective for eliminating dimensional discrepancies, accelerating convergence in machine learning algorithms, and enhancing the performance of distance-based algorithms such as  $k$ -means clustering and  $k$ -nearest neighbors. It is especially useful when feature values vary significantly and a comprehensive comparison across features is required.

To evaluate the effectiveness of our proposed feature selection algorithm, we selected six other feature selection algorithms for comparison. These include two single-granularity rough set based on granular-balls (GBRS [21], GBNRS [20]), two traditional multi-granularity rough set without granular-balls (GMDNRS [30], FNPME-FS [29]), one single-granularity rough set using zentropy as a metric (Ze-FS [34]), and one non-rough set feature selection algorithm (MEL [37]). Additionally, we also tested the classification effect without feature selection (Raw).

After feature selection, the processed data were classified using three classifiers, namely the Extra Trees Classifier (ETC [38]), Gradient Boosting Classifier (GBC [39]), and Support Vector Classifier (SVC [40]). All classifiers were implemented using the sklearn package with default parameters. Specifically, for the Extra Trees Classifier, the number of trees in the forest was set to 100; other parameter details can be found in the Extra Trees Classifier Documentation. For the Gradient Boosting Classifier, the learning rate was set to 0.1, and the number of boosting stages to perform was 100; additional parameters are described in the Gradient Boosting Classifier Documentation. For the Support Vector Classifier, the regularization parameter  $C$  was set to 1.0, the kernel function was set to rbf, and the polynomial degree was set to 3; further settings can be found in the Support Vector Classifier Documentation.

For the above algorithms, we set the  $\delta$  and  $\beta$  parameters of GMDNRS to range from 0.1 to 0.5 and 0.6 to 0.9, respectively, with a step size of 0.1. We set the  $\alpha$  parameter of FNPME-FS to range from 0.1 to 0.5, with a step size of 0.1, and also set  $\delta$  for FNPME-FS from 0.1 to 0.5, with a step size of 0.1. For our algorithm (GMG-GBRS), we set the minimum and maximum size of multiple granular space to 100 and 5000, and a minimum granular-ball size of 2. The value of  $\beta$  was varied from 0.6 to 1, with a step size of 0.1. The maximum granular-ball size was set as 0.05, 0.1, 0.2, 0.3, 0.4, and 0.5 times the sample size. Other algorithms do not need to set parameters.

It is important to note that considering the multiple granular space for GMDNRS and FNPME-FS is extremely time-consuming, given that their size is  $2^{|A|}$ . Therefore, in this subsection, we have limited the size of the multiple granular space calculated by these two algorithms by setting the maximum size of multiple granular space to 100, which is substantially lower than that of GMG-GBRS, which is 5000. Setting the maximum size of multiple granular space to 5000 for GMDNRS and FNPME-FS would result in excessive computation time and make it impractical to obtain results.

For all experiments, we performed 10-fold cross-validation and calculated the mean and standard deviation of the results.

The performance of each feature selection algorithm across different datasets, using the three classifiers, is presented in Table 3, Table 4, and Table 5. The number of selected features is shown in Table 6.

As evident from the results, the algorithm we proposed demonstrates outstanding performance across all classifiers. Additionally, it effectively extracts features and simplifies the attributes.

**Table 3**  
Classification Accuracy and Its Standard Deviation (%) with Extra Trees Classifier.

	Raw	GBRS	GBNRS	GMDNRS	FNPME-FS	Ze-FS	MEL	GMG-GBRS
Cancer	74.39±14.44	77.95±17.06	75.91±12.22	72.50±14.83	77.73±10.39	77.73±10.39	42.42±16.01	<b>78.64±12.46</b>
Darwin	89.71±7.46	90.33±8.74	90.82±4.89	89.71±7.46	90.29±5.91	67.35±14.99	89.12±5.65	<b>91.44±6.63</b>
Wine	<b>98.89±2.34</b>	97.78±3.88	<b>98.89±2.34</b>	78.76±8.88	<b>98.89±2.34</b>	<b>98.89±2.34</b>	97.75±3.92	<b>98.89±2.34</b>
Sonar	85.07±8.29	82.67±7.27	88.40±9.95	82.64±6.98	84.12±9.88	61.55±7.70	73.95±11.19	<b>90.88±6.89</b>
Heart1	73.84±5.36	70.72±4.40	74.49±2.81	61.91±3.39	74.56±6.50	73.86±7.65	74.13±4.46	<b>75.47±7.76</b>
Ionosphere	94.01±3.92	94.29±3.31	94.29±3.02	92.29±5.06	94.58±3.43	87.46±6.49	92.30±3.32	<b>94.59±2.84</b>
Urban	86.38±3.46	86.67±1.71	87.84±3.64	86.38±3.46	86.66±3.72	67.43±4.26	86.37±3.61	<b>87.99±3.26</b>
Qsar	87.49±3.13	82.94±2.66	87.87±2.69	86.35±2.38	87.02±2.50	86.06±3.00	85.97±2.81	<b>87.87±2.48</b>
Svmguide3	85.28±1.70	82.09±1.38	<b>85.83±1.79</b>	82.55±2.92	85.51±2.00	85.51±2.21	73.75±0.33	85.04±2.44
Semeion	94.10±1.40	89.71±1.45	94.85±1.60	94.10±1.40	<b>95.04±1.27</b>	63.72±5.73	75.27±2.54	94.91±1.84
Tunadromd	<b>99.57±0.22</b>	99.53±0.25	<b>99.57±0.22</b>	<b>99.57±0.22</b>	98.61±0.41	98.95±0.40	98.90±0.39	99.53±0.29
Spambase	95.72±0.74	90.89±1.17	<b>95.94±0.71</b>	78.50±2.26	94.20±0.83	94.57±1.13	93.61±0.81	95.72±0.74
Quality	69.93±1.55	70.13±2.28	<b>70.15±1.66</b>	64.54±2.23	57.08±1.43	70.13±1.68	64.23±1.49	69.93±1.55
Mushroom	<b>100.00±0.00</b>	<b>100.00±0.00</b>	<b>100.00±0.00</b>	98.21±0.63	<b>100.00±0.00</b>	<b>100.00±0.00</b>	99.89±0.10	<b>100.00±0.00</b>
Htru3	97.53±0.59	97.48±0.66	97.58±0.59	92.17±0.90	96.85±0.68	97.53±0.59	94.99±0.90	<b>97.63±0.45</b>
Mushroom1	<b>100.00±0.00</b>	<b>100.00±0.00</b>	<b>100.00±0.00</b>	99.21±0.19	<b>100.00±0.00</b>	<b>100.00±0.00</b>	98.57±0.37	<b>100.00±0.00</b>
Bean	92.25±0.95	91.91±0.83	92.23±0.86	92.44±0.72	91.29±0.74	<b>92.79±0.67</b>	92.76±0.82	92.43±0.84
Telescope	<b>87.85±0.63</b>	87.40±0.64	87.52±0.60	74.58±1.19	61.62±0.92	<b>87.85±0.63</b>	83.20±0.78	<b>87.85±0.63</b>

**Table 4**  
Classification Accuracy and Its Standard Deviation (%) with Gradient Boosting Classifier.

	Raw	GBRS	GBNRS	GMDNRS	FNPME-FS	Ze-FS	MEL	GMG-GBRS
Cancer	77.05±14.44	77.88±13.45	77.05±14.44	75.83±12.94	76.89±10.95	77.05±14.44	40.83±19.55	<b>79.47±9.68</b>
Darwin	87.52±8.48	87.55±9.28	88.59±6.97	87.52±8.48	84.61±9.09	61.63±10.65	87.42±6.80	<b>89.31±11.19</b>
Wine	93.89±7.15	<b>97.22±2.93</b>	95.56±6.31	77.48±9.47	95.00±7.15	95.00±6.11	95.52±6.84	96.63±3.91
Sonar	85.14±9.07	81.76±12.02	84.64±8.36	82.17±11.02	83.17±12.88	56.12±12.80	75.98±8.11	<b>85.19±10.13</b>
Heart1	73.46±4.80	70.34±5.85	74.14±4.05	61.91±3.39	74.90±6.51	74.55±7.29	74.46±5.22	<b>74.90±5.89</b>
Ionosphere	94.00±3.92	92.87±3.88	93.72±3.25	90.59±4.49	94.00±4.56	88.33±7.15	91.44±4.87	<b>94.02±3.15</b>
Urban	88.15±3.70	88.31±3.21	88.30±2.73	88.15±3.70	87.86±3.71	67.88±4.91	87.27±3.19	<b>88.89±3.30</b>
Qsar	85.69±2.78	82.94±3.41	86.45±2.60	85.97±3.20	86.35±2.32	85.59±1.52	86.08±3.53	<b>87.30±3.30</b>
Svmguide3	85.44±2.01	82.40±1.73	<b>86.29±1.63</b>	79.98±2.85	85.12±1.90	86.06±1.73	73.75±0.33	85.74±2.54
Semeion	92.72±2.11	90.27±1.48	93.16±1.93	92.72±2.11	92.84±1.70	65.35±3.67	74.64±3.79	<b>93.22±1.42</b>
Tunadromd	98.72±0.48	98.16±0.55	<b>98.84±0.41</b>	98.72±0.48	97.49±0.41	97.45±0.78	98.07±0.50	98.75±0.46
Spambase	94.50±0.84	88.79±1.41	<b>94.72±1.00</b>	79.68±1.68	92.70±1.16	93.46±0.90	93.02±0.97	94.50±0.84
Quality	59.13±1.38	59.13±1.71	<b>59.47±1.81</b>	52.00±1.80	48.92±1.84	59.39±1.67	54.98±1.15	59.13±1.38
Mushroom	<b>100.00±0.00</b>	<b>100.00±0.00</b>	<b>100.00±0.00</b>	98.04±0.58	<b>100.00±0.00</b>	99.62±0.22	99.89±0.10	<b>100.00±0.00</b>
Htru3	97.25±0.69	97.27±0.73	<b>97.30±0.60</b>	93.15±1.09	96.93±0.74	97.25±0.69	96.53±0.87	97.30±0.68
Mushroom1	99.98±0.08	<b>100.00±0.00</b>	<b>100.00±0.00</b>	98.88±0.33	98.83±0.19	<b>100.00±0.00</b>	98.57±0.37	<b>100.00±0.00</b>
Bean	92.59±0.61	92.01±0.87	92.61±0.61	92.53±0.82	91.24±0.82	<b>92.65±0.65</b>	92.33±0.80	92.59±0.72
Telescope	<b>87.16±0.62</b>	86.75±0.47	87.05±0.53	74.64±0.54	66.70±0.46	<b>87.16±0.62</b>	84.06±0.56	<b>87.16±0.62</b>

**Table 5**  
Classification Accuracy and Its Standard Deviation (%) with Support Vector Classifier.

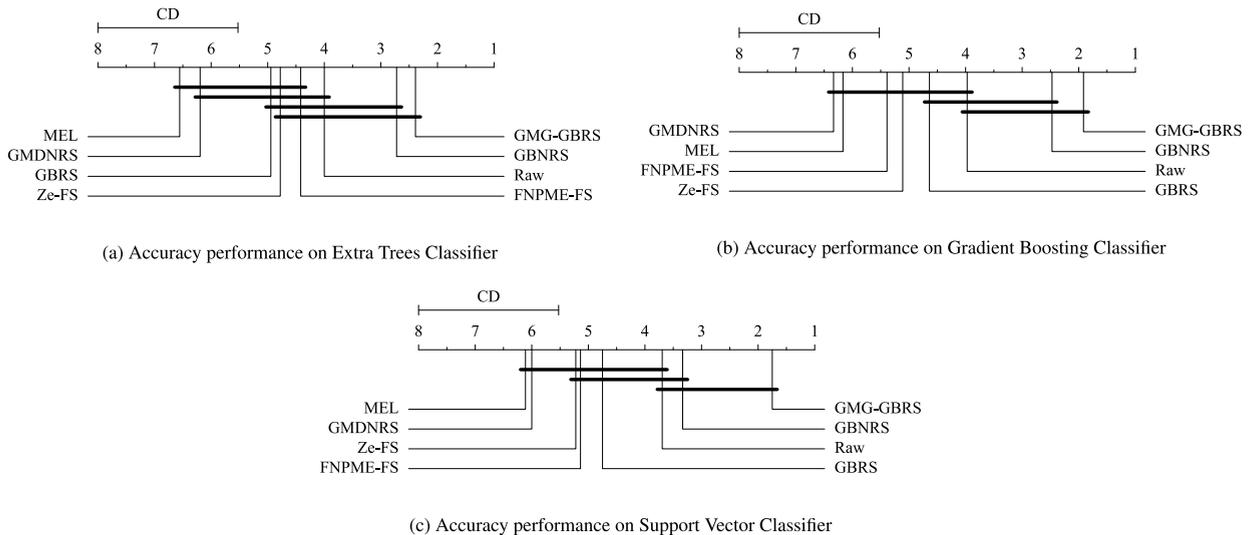
	Raw	GBRS	GBNRS	GMDNRS	FNPME-FS	Ze-FS	MEL	GMG-GBRS
Cancer	72.65±14.97	<b>78.71±12.43</b>	73.48±13.07	78.56±12.23	75.98±12.21	75.98±12.21	61.36±16.76	78.71±16.20
Darwin	86.86±7.49	86.86±7.49	86.86±8.09	86.86±7.49	86.24±8.21	68.46±11.32	83.99±7.96	<b>88.63±9.45</b>
Wine	98.33±3.75	97.78±2.87	<b>98.33±2.68</b>	82.12±9.27	98.33±3.75	98.33±3.75	97.78±3.88	<b>98.33±2.68</b>
Sonar	84.10±10.23	81.21±10.72	84.57±11.57	79.19±9.79	79.79±11.71	66.38±11.13	76.93±12.77	<b>86.00±9.87</b>
Heart1	74.85±6.29	74.55±6.94	76.89±6.30	61.91±3.39	75.25±6.81	74.55±6.94	74.52±4.37	<b>78.87±4.36</b>
Ionosphere	93.15±3.62	94.58±3.92	93.72±3.52	92.87±3.38	93.72±3.25	90.30±6.90	91.45±3.56	<b>94.58±2.51</b>
Urban	84.30±1.67	84.30±1.67	84.31±2.15	84.30±1.67	83.85±1.89	63.27±4.59	<b>84.60±3.70</b>	84.46±2.55
Qsar	85.41±2.81	78.49±3.11	<b>86.16±3.13</b>	83.90±3.96	85.41±2.22	83.42±3.11	84.27±2.66	85.79±1.86
Svmguide3	79.75±1.60	78.58±1.76	80.06±1.78	76.72±1.70	80.45±1.87	<b>81.07±1.53</b>	73.75±0.33	80.53±1.17
Semeion	95.60±1.60	91.40±1.12	<b>95.61±1.45</b>	95.60±1.60	95.48±1.65	65.47±4.99	74.90±2.55	95.17±1.54
Tunadromd	98.77±0.28	98.88±0.52	98.81±0.32	98.77±0.28	98.07±0.40	98.19±0.58	98.34±0.40	<b>98.97±0.19</b>
Spambase	<b>93.31±0.75</b>	86.20±1.37	93.26±0.71	79.22±1.53	90.65±1.12	91.24±1.31	91.11±1.02	<b>93.31±0.75</b>
Quality	55.06±1.35	54.84±1.01	55.06±1.35	47.49±0.88	45.24±1.27	<b>55.10±1.32</b>	53.23±1.66	55.06±1.35
Mushroom	<b>100.00±0.00</b>	99.93±0.17	98.96±0.40	96.10±0.98	99.89±0.15	97.86±0.33	99.59±0.16	<b>100.00±0.00</b>
Htru3	97.27±0.68	97.27±0.75	97.27±0.68	93.30±0.68	97.18±0.74	97.30±0.67	96.98±0.71	<b>97.33±0.74</b>
Mushroom1	<b>100.00±0.00</b>	99.63±0.22	99.46±0.25	98.97±0.23	<b>100.00±0.00</b>	99.90±0.13	98.55±0.40	<b>100.00±0.00</b>
Bean	92.44±0.80	92.01±0.54	92.54±0.76	92.49±0.73	90.79±0.81	92.48±0.78	92.52±0.76	<b>92.62±0.74</b>
Telescope	<b>85.98±0.61</b>	85.02±0.60	85.69±0.62	72.88±0.63	66.49±0.37	<b>85.98±0.61</b>	81.69±0.84	<b>85.98±0.61</b>

**Table 6**  
The Number of Selected Features.

	Raw	GBRS			GBNRS			GMDNRS			FNPME-FS			Ze-FS			MEL			GMG-GBRS		
	-	ETC	GBC	SVC	ETC	GBC	SVC	ETC	GBC	SVC	ETC	GBC	SVC	ETC	GBC	SVC	ETC	GBC	SVC	ETC	GBC	SVC
Cancer	9	7	7	5	6	9	5	3	3	3	6	6	6	6	9	6	2	2	2	4	4	8
Darwin	450	449	449	449	319	319	319	450	450	450	327	327	327	2	2	2	172	172	172	255	245	211
Wine	13	8	9	9	11	10	10	2	2	2	9	9	13	13	9	13	5	5	5	12	9	12
Sonar	60	25	23	10	57	58	54	15	15	15	33	33	33	2	2	2	3	3	3	46	32	52
Heart1	13	8	8	4	11	10	10	3	3	3	2	2	2	4	4	4	7	7	7	5	4	4
Ionosphere	34	15	11	15	32	13	29	11	6	6	30	31	30	4	4	4	7	7	7	26	22	26
Urban	147	145	145	145	142	141	130	147	147	147	124	138	138	9	9	9	41	41	41	56	54	55
Qsar	41	8	5	5	38	37	38	20	20	20	31	10	31	20	20	14	14	14	26	30	30	
Svmguide3	22	9	9	9	19	21	19	5	5	5	17	17	17	21	20	20	2	2	2	11	8	8
Semeion	256	76	76	76	250	241	245	256	256	256	241	249	250	7	7	7	35	35	35	46	45	45
Tunadromd	241	187	187	187	119	183	183	241	241	241	81	81	81	27	27	27	56	56	56	29	59	33
Spambase	57	15	15	15	56	52	56	2	2	2	24	24	24	35	39	39	20	20	20	57	57	57
Quality	11	10	10	10	9	9	11	4	4	4	2	2	2	10	10	10	3	3	3	11	11	11
Mushroom	22	20	20	20	6	6	9	5	5	5	12	18	18	9	9	9	4	4	4	22	22	22
Htru3	8	4	7	7	7	7	8	2	2	2	2	2	2	8	8	7	2	2	2	5	4	4
Mushroom1	22	12	12	19	5	5	5	11	11	11	10	10	10	10	10	13	2	2	2	12	17	17
Bean	16	8	8	8	13	15	13	13	13	13	8	8	8	7	7	7	5	5	5	11	11	12
Telesope	10	8	8	9	8	8	9	3	3	2	2	2	2	10	10	10	3	3	3	10	10	10

**Table 7**  
Statistical Test of Feature Selection Algorithms.

	Raw	GBRS	GBNRS	GMDNRS	FNPME-FS	Ze-FS	MEL	GMG-GBRS	P-Value	CD
ETC	4.00	4.94	2.72	6.19	4.42	4.78	6.56	2.39	$2.12 \times 10^{-8}$	2.47
GBC	3.97	4.64	2.47	6.33	5.39	5.11	6.17	1.92	$6.40 \times 10^{-10}$	2.47
SVC	3.69	4.75	3.33	6.00	5.14	5.22	6.11	1.75	$5.03 \times 10^{-8}$	2.47



**Fig. 6.** Accuracy performance on different classifiers.

To assess whether there is a statistical difference in classification performance, the Friedman test was conducted at a significance level of  $P = 0.05$ . Table 7 presents the average rankings of the eight methods along with the results of the Friedman test. The p-values are all significantly smaller than 0.05, indicating that there are significant differences among the eight methods for the three classifiers. Therefore, the Nemenyi post hoc test was performed to further investigate if there are substantial differences between any three methods. In the Nemenyi test, the critical distance (CD) is calculated as follows:

$$CD = q_{\alpha} \sqrt{\frac{k(k+1)}{6N}}, \tag{31}$$

where  $q_{0.05} = 3.031$  when  $k = 8$  and  $N = 18$ . A significant difference is considered to exist when the distance between two compared methods exceeds the critical distance ( $CD \approx 2.47$ ).

**Table 8**  
Runtime (s) for Computing Uncertainty Measures in Multi-Granularity Rough Set-Based Feature Selection.

No.	Dataset	GMDNRS	FNPME-FS	GMG-GBRS	No.	Dataset	GMDNRS	FNPME-FS	GMG-GBRS
1	Cancer	0.24	0.10	1.34	10	Semeion	3097.27	1604.07	157.99
2	Darwin	72.69	31.14	28.84	11	Tunadromd	23598.12	17721.29	365.88
3	Wine	4.07	1.39	4.17	12	Spambase	7191.95	3068.37	108.85
4	Sonar	18.73	5.40	13.11	13	Quality	1095.20	549.72	54.95
5	Heart1	6.94	3.66	8.42	14	Mushroom	8040.25	3119.56	64.44
6	Ionosphere	23.99	10.75	13.87	15	Htru3	269.17	134.27	53.17
7	Urban	286.56	161.87	44.21	16	Mushroom1	9505.26	3663.98	102.62
8	Qsar	224.01	89.93	23.11	17	Bean	16066.28	10010.58	534.16
9	Svmguide3	268.31	112.56	17.54	18	Telescope	6159.78	3786.78	486.99

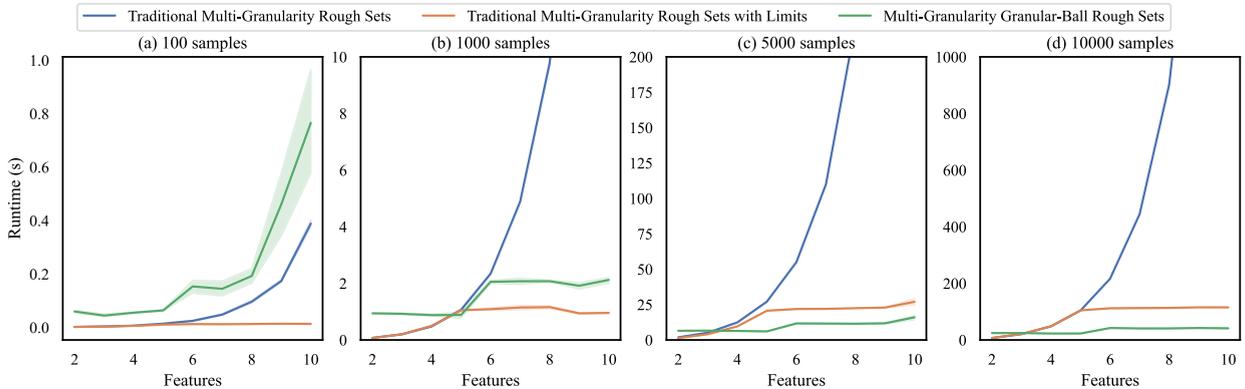


Fig. 7. Runtime (s) of Computing Lower Approximations.

The results of the Nemenyi test are presented in Fig. 6a, Fig. 6b, and Fig. 6c, which show the critical distance diagrams reflecting the rankings of the eight methods. The methods with lower ranks perform better. As shown in these figures, GMG-GBRS ranks first across all metrics and is statistically superior to the other compared methods in most cases.

### 5.3. Efficiency of GMG-GBRS

To ensure a fair and meaningful comparison, we exclusively select baseline algorithms that are based on traditional multi-granularity rough set (MGRS) frameworks. This is because our proposed multi-granularity granular-balls is specifically designed to enhance the multi-granularity rough set model, and comparing it with fundamentally different approaches would obscure the advantages it brings to this particular class of algorithms.

Compared to traditional MGRS methods, our proposed generalized multi-granularity granular-balls rough set (GMG-GBRS) not only achieves superior feature selection performance, but also demonstrates significantly improved computational efficiency. As shown in Table 8, the time required to compute uncertainty measures with GMG-GBRS is markedly lower than that of GMDNRS and FNPME-FS—two representative MGRS-based methods that do not leverage multi-granularity granular-balls. This substantial reduction in computation time greatly accelerates the overall feature selection process.

It is noteworthy that, in contrast to the findings presented in Table 3, Table 4, and Table 5, the maximum size of multiple granular space for GMDNRS and FNPME-FS in Table 8 has been set to 5000, aligning with the setting used for GMG-GBRS, rather than 100. This adjustment was made to facilitate the comparison of results under uniform conditions and to uphold the rigor of the derived conclusions.

To explain the high efficiency of GMG-GBRS, we tested it on datasets with 100, 1000, 5000, and 10,000 samples. As shown in Fig. 7, we measured the time required for the traditional multi-granularity rough set to compute the multi-granularity lower approximation across all granularities, the size of which is the cardinality of the power set of the attribute set. Additionally, we tested the time taken when the multiple granular space was restricted to 32.

From Fig. 7, we observe that when the multiple granular space of the traditional multi-granularity rough set is unrestricted, the running time grows exponentially with the number of features. However, when the multiple granular space is limited to 32, the running time becomes primarily dependent on the number of samples. We then used GMG-GBRS to calculate the lower approximation under the same conditions, with the minimum size of multiple granular space set to 32 and no restriction on the maximum size of multiple granular space. The results show that the running time of GMG-GBRS grows similarly to the traditional rough set when the multiple granular space is restricted, but since our algorithm screens the granularity during the granular-balls generation process, it avoids considering certain granularities when computing the lower approximation, thereby suppressing the exponential growth in running time as the number of features increases.

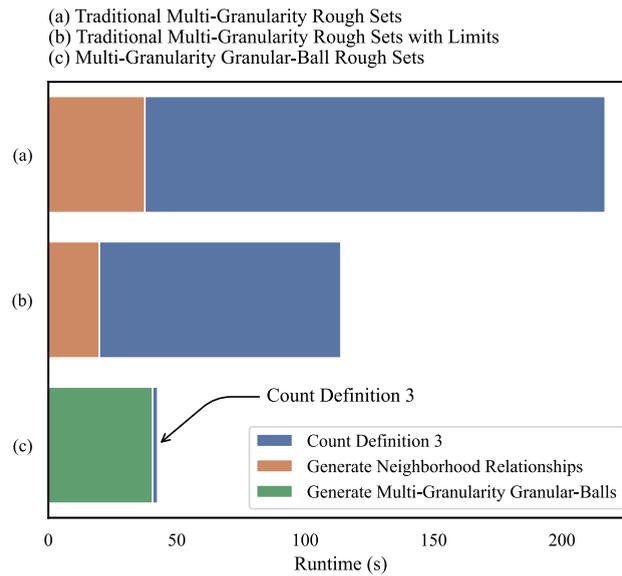


Fig. 8. The Time Required (s) Each Step of Algorithms.

Next, we analyzed the time required for each step of the algorithms when calculating the lower approximation with 10,000 samples and 6 features, as shown in Fig. 8. We found that for the traditional multi-granularity rough set algorithm without limiting the multiple granular space, most of the time was spent calculating the lower approximation (as defined in Definition 3). This trend also applies when the multiple granular space is restricted. However, for GMG-GBRS, most of the computing time is spent generating multi-granularity granular-balls, and the time spent calculating the lower approximation is relatively small. This is because after generating the granular-balls, our domain size is reduced from the sample size  $|U|$  to the number of multi-granularity granular-balls  $|\mathcal{G}\mathcal{B}|$ , which significantly speeds up the computation of the lower approximation.

These two findings demonstrate that our algorithm effectively overcomes the limitations of traditional multi-granularity rough set and enables efficient feature selection based on multi-granularity rough set.

### 6. Summary and future research directions

In this paper, we propose an algorithm that advances granular-balls generation and enables effective multi-granularity modeling. Our contributions are threefold. First, we introduce a neighbor search-based granular-balls generation algorithm that incrementally forms granular-balls by grouping each data point with its local neighbors. This method is agnostic to data distribution and supports more accurate modeling of local neighborhoods. Second, we extend this approach to define and construct multi-granularity granular-balls, which facilitate data representation across multiple granular levels. By filtering out invalid granularities during the generation process, our method significantly reduces granularity complexity while accelerating downstream tasks such as feature selection. Third, we develop a feature selection algorithm grounded in generalized multi-granularity granular-ball rough set, incorporating Zentropy [34,35] as a robust uncertainty measure. This framework jointly reduces the domain size and the granularity search space, thereby improving computational efficiency without sacrificing selection accuracy.

In the experiments presented in Section 5, our algorithm demonstrated strong performance in both feature selection effectiveness and computational efficiency. We attribute these results to the following key advantages and their underlying reasons:

1. By leveraging neighbor-based generation, our method constructs granular-balls even in datasets with arbitrary or irregular distributions, enabling more accurate neighborhood relations and tighter approximations. When combined with Zentropy, it allows for more precise uncertainty quantification across different granular levels.
2. Unlike conventional multi-granularity rough set that require exhaustive exploration of the power set of attributes, our algorithm selectively filters out ineffective granularities during granular-ball generation, thereby reducing the computational burden and improving overall efficiency.
3. In line with the original granular-balls, our method compresses the universe from the full sample space to the space of granular-balls, significantly accelerating rough set computations across all levels of granularity.

Despite the promising performance of the proposed method, several important challenges remain to be addressed:

1. Although our approach improves the efficiency of multi-granularity granular-ball rough set, it still faces difficulties when applied to ultra-large-scale or continuously evolving (incremental) datasets. These scenarios, commonly encountered in real-world applications such as streaming data analysis, demand more lightweight, scalable, and adaptive solutions.

2. The quality and speed of granular-ball generation—especially in establishing neighborhood relationships—directly influence the effectiveness of subsequent tasks like feature selection. Developing faster and more accurate construction strategies remains an open challenge, particularly for high-dimensional or noisy datasets.
3. Although Zentropy is effective for multi-granularity uncertainty estimation, further research is needed to design more expressive or task-specific entropy-based measures that can better capture structural information across diverse granularities.

These open issues suggest several promising directions for future research:

1. Develop incremental algorithms for both granular-ball generation and multi-granularity rough set computation, enabling real-time adaptation to streaming or dynamically changing data without requiring full recomputation.
2. Investigate novel granular-ball generation strategies beyond neighbor-based methods. In parallel, explore advanced entropy models that support finer-grained and task-aware uncertainty assessment.
3. Expand the application of multi-granularity granular-balls to other machine learning problems—such as classification, clustering, and anomaly detection—building on their demonstrated effectiveness in feature selection to enhance both model performance and interpretability within granular computing frameworks.

In summary, our work bridges the methodological gap between granular-ball rough set and multi-granularity modeling by introducing a unified, efficient, and robust framework. Beyond enhancing the performance of feature selection, it lays a solid foundation for future research and broader applications in granular computing and data mining.

### CRedit authorship contribution statement

**Weirui Ye:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Weihua Xu:** Validation, Supervision, Project administration, Methodology, Investigation, Funding acquisition, Conceptualization.

### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

### Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grant 62376229 and in part by the Natural Science Foundation of Chongqing under Grant CSTB2023NSCQLZX0027.

### Data availability

No data was used for the research described in the article.

### References

- [1] Z. Pawlak, Rough sets, *Int. J. Comput. Inf. Sci.* 11 (1982) 341–356.
- [2] M.A.N.D. Sewwandi, Y. Li, J. Zhang, A class-specific feature selection and classification approach using neighborhood rough set and k-nearest neighbor theories, *Appl. Soft Comput.* 143 (2023) 110366.
- [3] B. Zhang, L. Zhang, *Theory and Applications of Problem Solving*, New Comprehensive Biochemistry, North Holland, 1992.
- [4] D. Li, Membership clouds and membership cloud generators, *Comput. Res. Dev.* 32 (6) (1995) 15–20.
- [5] X. Zhang, Y. Huang, Optimal scale selection and knowledge discovery in generalized multi-scale decision tables, *Int. J. Approx. Reason.* 161 (2023) 108983.
- [6] W. Xu, K. Cai, D.D. Wang, A novel information fusion method using improved entropy measure in multi-source incomplete interval-valued datasets, *Int. J. Approx. Reason.* 164 (2024) 109081.
- [7] Z. Yuan, B. Chen, J. Liu, H. Chen, D. Peng, P. Li, Anomaly detection based on weighted fuzzy-rough density, *Appl. Soft Comput.* 134 (2023) 109995.
- [8] L. Ou-Yang, X.-F. Zhang, X.-M. Zhao, D.D. Wang, F.L. Wang, B. Lei, H. Yan, Joint learning of multiple differential networks with latent variables, *IEEE Trans. Cybern.* 49 (9) (2018) 3494–3506.
- [9] D.D. Wang, H. Xie, H. Yan, Proteo-chemometrics interaction fingerprints of protein–ligand complexes predict binding affinity, *Bioinformatics* 37 (17) (2021) 2570–2579.
- [10] L. Chen, Topological structure in visual perception, *Science* 218 (4573) (1982) 699–700.
- [11] G. Wang, Dgccc: data-driven granular cognitive computing, *Granul. Comput.* 2 (4) (2017) 343–355.
- [12] S. Xia, Y. Liu, X. Ding, G. Wang, H. Yu, Y. Luo, Granular ball computing classifiers for efficient, scalable and robust learning, *Inf. Sci.* 483 (2019) 136–152.
- [13] S. Xia, G. Wang, X. Gao, Granular ball computing: an efficient, robust, and interpretable adaptive multi-granularity representation and computation method, *arXiv preprint, arXiv:2304.11171*, 2023, <https://doi.org/10.48550/arXiv.2304.11171>.
- [14] H. Zhao, D.D. Wang, L. Chen, X. Liu, H. Yan, Identifying multi-dimensional co-clusters in tensors based on hyperplane detection in singular vector spaces, *PLoS ONE* 11 (9) (2016) e0162293.
- [15] S. Xia, X. Lian, G. Wang, X. Gao, J. Chen, X. Peng, Gbsvm: an efficient and robust support vector machine framework via granular-ball computing, *IEEE Trans. Neural Netw. Learn. Syst.* 36 (5) (2025) 9253–9267.
- [16] Y. Xue, Y. Shao, S. Xia, G. Wang, The dual model of support vector machine based on granular ball computing, in: *2021 3rd International Conference on Applied Machine Learning (ICAML)*, IEEE, 2021, pp. 43–47.

- [17] Y. Shao, Y. Hua, Z. Gong, X. Zhu, Y. Cheng, L. Li, S. Xia, Con-mgsvm: controllable multi-granularity support vector algorithm for classification and regression, *Inf. Fusion* 117 (2025) 102867.
- [18] S. Xia, D. Peng, D. Meng, C. Zhang, G. Wang, E. Giem, W. Wei, Z. Chen, Ball k-means: fast adaptive clustering with no bounds, *IEEE Trans. Pattern Anal. Mach. Intell.* 44 (1) (2020) 87–99.
- [19] J. Xie, W. Kong, S. Xia, G. Wang, X. Gao, An efficient spectral clustering algorithm based on granular-ball, *IEEE Trans. Knowl. Data Eng.* 35 (9) (2023) 9743–9753.
- [20] S. Xia, H. Zhang, W. Li, G. Wang, E. Giem, Z. Chen, Gbnrs: a novel rough set algorithm for fast adaptive attribute reduction in classification, *IEEE Trans. Knowl. Data Eng.* 34 (3) (2020) 1231–1242.
- [21] S. Xia, C. Wang, G. Wang, X. Gao, W. Ding, J. Yu, Y. Zhai, Z. Chen, Gbrs: a unified granular-ball learning model of Pawlak rough set and neighborhood rough set, *IEEE Trans. Neural Netw. Learn. Syst.* 36 (1) (2025) 1719–1733.
- [22] W. Qian, F. Xu, J. Qian, W. Shu, W. Ding, Multi-label feature selection based on rough granular-ball and label distribution, *Inf. Sci.* 650 (2023) 119698.
- [23] W. Shu, Y. Hu, W. Qian, Multi-label feature selection for missing labels by granular-ball based mutual information, *Appl. Intell.* 54 (2024) 12589–12612.
- [24] W. Qian, F. Xu, J. Huang, J. Qian, A novel granular ball computing-based fuzzy rough set for feature selection in label distribution learning, *Knowl.-Based Syst.* 278 (2023) 110898.
- [25] Y. Sun, P. Zhu, Online group streaming feature selection based on fuzzy neighborhood granular ball rough sets, *Expert Syst. Appl.* 249 (2024) 123778.
- [26] S. Xia, L. Xiaoyu, B. Sang, G. Wang, X. Gao, Gbfrs: robust fuzzy rough sets via granular-ball computing, *arXiv preprint, arXiv:2501.18413*, 2025, <https://doi.org/10.48550/arXiv.2501.18413>.
- [27] W. Xu, Y. Zhang, Y. Qian, A novel unsupervised feature selection for high-dimensional data based on fcm and  $k$ -nearest neighbor rough sets, *IEEE Trans. Neural Netw. Learn. Syst.* (2024), <https://doi.org/10.1109/TNNLS.2024.3460796>.
- [28] W. Xu, W. Ye, Incremental feature selection: parallel approach with local neighborhood rough sets and composite entropy, *Pattern Recognit.* 159 (2025) 111141.
- [29] L. Sun, L. Wang, W. Ding, Y. Qian, J. Xu, Feature selection using fuzzy neighborhood entropy-based uncertainty measures for fuzzy neighborhood multigranulation rough sets, *IEEE Trans. Fuzzy Syst.* 29 (1) (2020) 19–33.
- [30] W. Xu, Q. Bu, Feature selection using generalized multi-granulation dominance neighborhood rough set based on weight partition, *IEEE Trans. Emerg. Top. Comput. Intell.* 9 (1) (2025) 213–227.
- [31] X. Zhang, W. Zhao, Uncertainty measures and feature selection based on composite entropy for generalized multigranulation fuzzy neighborhood rough set, *Fuzzy Sets Syst.* 486 (2024) 108971.
- [32] J. Xu, C. Zhou, S. Xu, L. Zhang, Z. Han, Feature selection based on multi-perspective entropy of mixing uncertainty measure in variable-granularity rough set, *Appl. Intell.* 54 (1) (2024) 147–168.
- [33] Y. Yang, D. Chen, Z. Ji, X. Zhang, L. Dong, A two-way accelerator for feature selection using a monotonic fuzzy conditional entropy, *Fuzzy Sets Syst.* 483 (2024) 108916.
- [34] K. Yuan, D. Miao, Y. Yao, H. Zhang, X. Zhao, Feature selection using zentropy-based uncertainty measure, *IEEE Trans. Fuzzy Syst.* 32 (4) (2024) 2246–2260.
- [35] K. Yuan, D. Miao, W. Pedrycz, W. Ding, H. Zhang, Ze-HFS: zentropy-based uncertainty measure for heterogeneous feature selection and knowledge discovery, *IEEE Trans. Knowl. Data Eng.* 36 (11) (2024) 7326–7339.
- [36] Q. Xie, Q. Zhang, S. Xia, F. Zhao, C. Wu, G. Wang, W. Ding, Gbg++: a fast and stable granular ball generation method for classification, *IEEE Trans. Emerg. Top. Comput. Intell.* 8 (2) (2024) 2022–2036.
- [37] X. Wang, H. Shanguan, F. Huang, S. Wu, W. Jia, Mel: efficient multi-task evolutionary learning for high-dimensional feature selection, *IEEE Trans. Knowl. Data Eng.* 36 (8) (2024) 4020–4033.
- [38] P. Geurts, D. Ernst, L. Wehenkel, Extremely randomized trees, *Mach. Learn.* 63 (2006) 3–42.
- [39] J.H. Friedman, Greedy function approximation: a gradient boosting machine, *Ann. Stat.* (2001) 1189–1232.
- [40] J. Platt, Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods, *Adv. Large Margin Classif.* 10 (3) (1999) 61–74.