Knowledge-Based Systems 262 (2023) 110257

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Matrix-based multi-granulation fusion approach for dynamic updating of knowledge in multi-source information

Xiaoyan Zhang, Xudong Huang *, Weihua Xu

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

ARTICLE INFO

Article history:
Received 18 August 2022
Received in revised form 27 December 2022
Accepted 2 January 2023
Available online 6 January 2023

Keywords:
Fixed aggregation operator
Matrix
Multi-source information system
Possible aggregation operator

ABSTRACT

Multisource information fusion is an important big data technology that plays a crucial role in the fields of data mining and knowledge discovery. Multigranulation information fusion is an effective technique for obtaining critical information from multisource data. However, obtaining meaningful information from a dynamic multisource information system using a multigranulation fusion strategy has rarely been researched. Therefore, this study explores a matrix-based dynamic updating strategy for multigranulation fusion operators. First, a matrix-based method for computing multigranular fusion operators is proposed. Second, the matrix-based dynamic updating mechanism of the multigranularity fusion operator is discussed and constructed for four cases (deleting objects, adding objects, deleting sources, and adding sources). Finally, four groups of dynamic algorithms and static methods are compared to verify the effectiveness of the dynamic algorithm. The experimental findings show that the matrix-dynamic updating multigranulation fusion operator approach is effective and efficient.

1. Introduction

Big Data have led to the proliferation of complicated information systems in the real world. Rough sets, established by Pawlak [1], have been widely used in various fields such as data analysis, knowledge processing [2,3], and image processing. Big Data can be used to rapidly identify and extract knowledge from various information systems. To address various problems, researchers have developed numerous expanded rough set models at various levels, including variable-precision rough sets [4], probabilistic rough sets [5,6], and composite rough sets [7,8]. By establishing an α-tolerance relationship, Leung et al. investigated a rough set approach for rule induction [9]. Du et al. achieved attribute reduction in information systems using feature-based dominance relations in incompletely ordered information systems [10]. Yang et al. created tolerance- and finite-tolerance-based multigranulation rough sets to manage incomplete information systems [11]. Many mature rough-set expansion approaches have been demonstrated to solve most data processing tasks. However, various rough-set approaches do not perform well in handling multi-source data, such as web [12] and multi-view data [13].

Among the many methods for processing multi-source data, the multigranulation decision theory is an effective method that can obtain useful data from multi-source data. Multigranularity methods process multisource data employing the concept of granulation to obtain important information. Using the concept of granulation, scholars have conducted various studies, as follows. Using multi-granulation decision theory, Qian et al. created a rough-set model to evaluate multi-source categorization data [14]. Xu et al. established a two-way learning approach for fuzzy datasets using information granules [15]. Zhang et al. proposed an adaptive multigranularity rough-set model that was conducive to building a framework for granular computing (Gc) information fusion [16]. Lin et al. developed an information fusion approach for a multi-source information system with uncertainty by integrating rough sets with evidence theory [17]. To investigate numerous fuzzy data sources, Lin et al. devised a fuzzy multigranulation information fusion approach [18]. Xu et al. proposed a fusion approach for converting multiple raw data from different sources into triangular fuzzy particles [19]. Utilizing conditional information entropy, Xu et al. developed a multi-source information fusion method [20]. Although the aforementioned methods can well acquire information from multisource data, these methods are insufficient in handling multisource data that change over time, or in other aspects. Changes in multisource data include changes in the object set, attribute set, feature value, etc. As an example, new observatories and devices have been introduced to gather new meteorological characteristic data to increase weather forecasting capabilities. Because the dynamic change in multisource data causes information granularity and a change in the knowledge structure, an effective and reliable real-time decision-making method must be established. Establishing

* Corresponding author.
E-mail addresses: zxy19790915@163.com (X.Y. Zhang), huangxdml@gmail.com (X.D. Huang), chxuwh@gmail.com (W.H. Xu).

https://doi.org/10.1016/j.knosys.2023.110257
0950-7051/

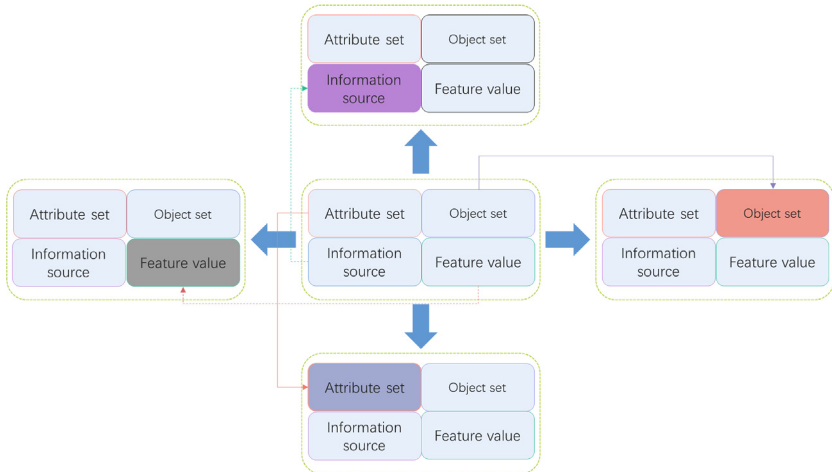

Fig. 1. The variation of single structural in a multi-source data.

a dynamic information acquisition method is conducive to acquiring knowledge and making decisions in real time based on dynamic data, and can reduce the time consumption incurred by repeated calculations.

Designing real-time computing models and efficient fusion algorithms for multisource dynamic information systems (MDIS) has gradually become a popular research topic. As an effective method for maintaining information granularity in dynamic information systems, incremental learning can combine original and new knowledge to obtain new information without recalculating the original information. A significant amount of research has been conducted on this subject. For example, Hu et al. implemented decision-making in dynamic situations by employing the matrix approach on the dynamic approximation updates of multigranulation rough sets [21]. Li et al. introduced a discovery technique based on incremental knowledge to analyze massive amounts of data with rich information using Gc and rough set theory as its foundations [22,23]. In addition, Zhang et al. introduced incremental information fusion methods for MDIS [24]. These methods are significant for obtaining information from dynamic data. In addition, from the Gc perspective, incremental learning methods primarily focus on the following four aspects (see Fig. 1) when handling dynamic multisource data.

(1) Dynamic information fusion strategies in the context of object variation. The dynamic data of the changes in the object set mainly include the addition or deletion of objects in the data. With increases and decreases in the object set, the information obtained from the data must be initially adjusted. The following are some important studies on handling dynamic data on object changes. Hu et al. introduced a matrix-based incremental technique for updating knowledge when granular structures are added or deleted [25]. Baszczyski et al. created an incremental method based on the Apriori algorithm to build effective decision rules and rule postprocessing technologies [26]. To update the approximation continually, Luo et al. devised two alternative incremental approaches [27]. Li et al. developed a technique for dynamically updating approximations in ordered information systems [28]. Li et al. demonstrated a technique for continuously improving the approximation by investigating the dominance relation inside the rough-set model [29]. For either adding and deleting numerous objects, Shu et al. created two selection strategies [30]. To achieve attribute reduction during active sample selection, Yang et al. proposed an incremental method based on rough set theory [31]. To maintain three areas in the multigranularity approximation space throughout the addition and removal of objects, Hu et al. created a dynamic framework based on a matrix [32].

(2) Dynamic information fusion strategies based on attribute modification. Changes in the dynamic data of the attribute set primarily include increases and decreases of the attributes in the data. A change in the attribute set causes a change in the data dimension and increases or decreases the amount of useful information. Chen et al. conducted research on updating the conceptual approximation while adding attributes and objects [33]. Wang et al. presented a dimensionality increment approach for reducing attribute sets while adding attribute sets [34]. Zhang et al. studied a method for dynamically updating approximations when deleting or inserting attributes in multigranularity interval hesitant fuzzy information systems [35]. Using a fuzzy rough-set model, Zeng et al. incrementally selected features in hybrid information systems [36]. Hu et al. suggested a technique for dynamically adjusting the three-way region of every decision category in an approximate set of probabilistic neighborhoods [37]. To avoid the reconstruction of the induced matrix in compound-order decision-making systems, Huang et al. demonstrated a dynamic technique based on matrix update attribute approximation [38].

(3) Approaches to dynamic information fusion with variable feature values. Dynamic data with changes in the feature values primarily cause the dynamic change in data through changes in specific values. Research on this aspect is relatively limited, primarily consisting of the following methods. Luo et al. obtained two approximate incremental algorithms when adding or deleting standard values [39]. Huang et al. proposed a matrix-based incremental update mechanism to dynamically update the approximate values of objects, attributes, and attribute values in multisource hybrid systems [40]. Jing et al. designed a technique for continuously updating the granularity of attribute reduction knowledge as the data values change [41]. Wei et al. investigated a reduction algorithm that incrementally obtains all reduction attributes of dynamic data [42].

(4) Approaches to information fusion that are dynamic based on the information source. Changes in the dynamic data of the information source change are primarily caused by changes in the information source. Researchers in this field have primarily focused on the information sources increasing and decreasing. Dubey et al. presented an intelligent wayfinding prediction framework for intersections with variable route selection under the influence of N directional information sources [43]. Huang et al. studied the incremental mechanism of the conditional entropy method using the matrix method and considered the dynamic update mechanism when deleting or adding data sources in multisource data [44]. By considering the simultaneous changes in attributes and information sources in a multisource interval value order information system, Xu et al. developed four dynamic update mechanisms for different situations [45]. Zhang et al. devised four dynamic multisource information fusion algorithms for recognizing attribute changes and merging data from different sources in the context of incomplete interval-valued datasets [46].

The series of dynamic models or algorithms proposed by scholars for the aforementioned four dynamic cases can be summarized as follows. Concerning methods for obtaining or updating information from the dynamic data of the object or attribute set changes, existing methods primarily include incremental learning and dynamic framework construction combined with a matrix. Studies on the changes of object sets primarily focuses on methods for dynamically updating the approximation space, decision rules, and attribute reduction. However, methods for updating information from multidynamic multisource data with feature value or information source changes are relatively more diverse. These two types of situations primarily depend on a need to build corresponding dynamic updating models and algorithms.

Although academics have published several strategies for processing dynamic data in different situations, few have explored the dynamic update of multigranulation fusion operators in multisource data. The fundamental goal of this study is to provide a matrix-based method for dynamically updating multigranulation fusion operators despite massive numbers of changes, such as the addition or removal of many objects or information sources. This study primarily expands on the following aspects. (1) We introduce the definitions of a decision vector, equivalence relation matrix, decision support matrix, decision related matrix, and fixed (possible) aggregation vector in terms of static multigranulation information fusion. (2) For adding or removing objects or information sources from multisource data, we present a matrix-based fusion technique that dynamically updates fixed (possible) aggregation operators. (3) The effectiveness of the proposed dynamic update multigranulation fusion operator algorithm was verified experimentally.

The main contributions of this study on dynamically updating multi-granulation fusion operators are as follows: (1) A method based on a matrix that can compute multigranulation fusion operators in multi-source information systems is proposed. (2) To dynamically update multigranulation information fusion operators, a matrix-based technique is presented to address the addition and removal of multiple objects and information sources. (3) The experimental findings show that the proposed dynamic multigranulation information fusion technique is more computationally efficient than the static approach.

The remainder of this paper is organized as follows. Section 2 introduces various notions of multisource decision information systems and multigranulation information fusion. In Section 3, we discuss a static multi-granulation information fusion approach based on a matrix. Section 4 presents a matrix-based technique for dynamically updating fixed and possible aggregation operators in a multisource decision information system when objects are added or removed. In Section 5, we explore dynamic approaches for updating fixed and possible aggregation operators in a multisource information system when sources are added or deleted. In Section 6, we compare the performances of static and dynamic information fusion techniques using a series of tests. Finally, Section 7 concludes the paper and presents directions for future work.

2. Preliminaries

This section provides the basis for the remainder of the paper by introducing various important concepts.

2.1. Fundamentals of the multigranulation rough-set model

Definition 2.1. Let an information system be denoted by $S = (U, AT, V, F)$, where $U = (x_1, x_2, \dots x_n)$, $AT = (a_1, a_2, \dots, a_n)$, $V = \bigcup_{a \in AT} V_a$, and $F = \{f \mid U \times AT \to V\}$, where U is the set of universe objects, AT the set of all attribute features in the information system, V the range of attribute values, and F an information function of the object and attribute sets [1].

Next, we introduce two important concepts for constructing multigranulation rough sets: equivalence relations and support characteristic functions. Simultaneously, two important multigranulation approximation operators are defined using these two basic concepts.

Definition 2.2. Let S represent an information system. A binary equivalence relation R_A is expressed as in [1]:

$$R_A = \left\{(x_i, x_j) \,\middle|\, f(x_i, a) = f(x_j, a)\right\} (x_i, x_j \in U, \forall a \in A).$$

Definition 2.3. The support characteristic function is defined as in [47]:

$$\varphi_{A_k}(x) = \begin{cases} 1, [x]_{A_k} \subseteq X \\ 0, otherwise \end{cases} \quad (A_k \subseteq AT(k \in \left\{1, 2, \dots, 2^{|AT|}\right\}),\ X \subseteq U), \tag{1}$$

where $[x]_{A_k}$ represents an equivalence class of object x under equivalence relation R_{A_k}.

Definition 2.4 ([47]). $\forall X \subseteq U$, multigranulation approximation operators $\underline{MG}_{\sum_{i=k}^{s} A_k}(X)_\beta$ and $\overline{MG}_{\sum_{i=k}^{s} A_k}(X)_\beta$ of X under relation $\sum_{i=k}^{s} A_k$ are expressed as follows:

$$\begin{aligned} \underline{MG}_{\sum_{i=k}^{s} A_k}(X)_\beta &= \left\{x \in U \,\middle|\, \frac{\sum_{k=1}^{s} \varphi_{A_k}(x)}{s} \geq \beta \right\}, \\ \overline{MG}_{\sum_{i=k}^{s} A_k}(X)_\beta &= \left\{x \in U \,\middle|\, \frac{\sum_{k=1}^{s} (1-\varphi_{A_k}(x))}{s} > 1 - \beta, \right\} \end{aligned} \tag{2}$$

where $\beta \in [0, 1]$, and $X \subseteq U$.

2.2. Multigranulation aggregation operator in multisource decision information system

A multi-source information system indicates that the data originates from numerous data sources. The following is a primary description of a multisource information system [48].

Definition 2.5. A multisource decision information system (MDIS) can be considered to comprise of multiple decision information systems (DISs), similar to $S_i = (U, AT, V_i, F_i, G)$, where U is the set of universe objects, $AT = C \cup D$ the set of all attribute features (including condition attributes C and decision attributes D) in the information system, V_i the range of condition attribute values in information source i, and G the range of decision attribute values. F_i is an information function between the object and attribute sets, representing the equivalence relationship of the conditional attribute set under information source i. The MDIS is often expressed as follows [42]:

$$MDIS = \{S_1, S_2, \dots, S_q\}. \tag{3}$$

Next, some basic concepts of multigranulation fusion methods in terms of multisource data are introduced, including two characteristic functions and two multigranulation fusion operators.

Definition 2.6. Given an *MDIS*, two characteristic functions are defined as follows [48]; the decision support and decision-related characteristic functions of x $\forall D_j \in U/D$ are defined as follows:

$$SC_{D_j}^{S_i}(x) = \begin{cases} 1, [x]_{S_i} \subseteq D_j \\ 0, otherwise \end{cases} (i \leq q), \tag{4}$$

$$RC_{D_j}^{S_i}(x) = \begin{cases} 1, [x]_{S_i} \cap D_j \neq \emptyset \\ 0, otherwise \end{cases} (i \leq q). \tag{5}$$

Here, $S_i(i = 1, 2, \ldots, q)$ denotes the ith decision information system, $[x]_{S_i}$ the equivalence class of object x under information source i, and D_j the jth class in division $U/D = \{D_1, D_2, \ldots, D_s\}$.

The two characteristic functions express different aspects, where $SC_{D_j}^{S_i}(x)$ indicates that object x will support decision class D_j in information source S_i, and $RC_{D_j}^{S_i}(x)$ indicates that object x may support decision class D_j in information source S_i.

Definition 2.7 (*[48]*)**.** Given an *MDIS*, (α, β) is a pair of thresholds. For any $D_j \in U/D$, the multi-granulation fusion operator of *MDIS* is formulated as follows, respectively:

$$\sum_{i=1}^{q} MS(D_j)_\alpha = \left\{ x \in U \left| \frac{\sum_{i=1}^{q} SC_{D_j}^{S_i}(x)}{q} \geq \alpha \right. \right\}, \tag{6}$$

$$\sum_{i=1}^{q} MR(D_j)_\beta = \left\{ x \in U \left| \frac{\sum_{i=1}^{q} RC_{D_j}^{S_i}(x)}{q} > \beta, \right. \right\} \tag{7}$$

where $\alpha + \beta = 1$ and $0 < \alpha \leq 1$.

3. Matrix-based representation of multi-granulation fusion operator

The multigranulation information fusion operator in the *MDIS* is the focus of this section. We provide a matrix-based fusion strategy. The matrix form of the two operators is provided, in addition to several fundamental concepts of multigranulation fusion approaches.

Definition 3.1. Let S be an integrated system. For any $x_i \in U(i = 1, 2, \ldots, n)$, $D_j \in U/D$, and the decision vector $G(D_j) = [g_{ij}]_{n\times 1}$ of D_j is denoted as

$$g_{ij} = \begin{cases} 1, if\, x_i \in D_j; \\ 0, otherwise \end{cases}. \tag{8}$$

Definition 3.2. Given an $MDIS = \{S_k|S_k = (U, AT, V_k, F_k), k = 1, 2, \ldots, q\}$, where $AT = C \cup D$, for any $c \in C$, the equivalence relation matrix $ME^k = [m^k{}_{ij}]_{n\times n}$ under information source S_k is defined as follows:

$$m^k{}_{ij} = \begin{cases} 1, if\ f(x_i, c) = f(x_j, c) \\ 0, otherwise \end{cases}. \tag{9}$$

In particular, the equivalence class $[x]_{S_i}$ of object x_i under source S_k can be represented by the ith column or ith row of equivalence relation matrix ME^k, that is, $[m_i^k]_{n\times 1}$ or $[m_i^k]_{1\times n}$. Next, two multigranulation fusion operators are constructed in terms of the multisource data through equivalence relation matrix ME^k and decision vector $G(D_j)$. In the multigranulation fusion approach, decision support matrix $S(D_j)$ and decision-related matrix $R(D_j)$ are formed by two operators as follows.

Definition 3.3. Given an $MDIS = \{S_k|S_k = (U, AT, V_k, F_k), k = 1, 2, \ldots, q\}$, for any $D_j \in U/D$, and decision vector $G(D_j)$ of D_j is denoted as $G(D_j) = [g_{ij}]_{n\times 1}$. The equivalence relation matrix ME^k of S_k is denoted as $ME^k = [m^k{}_{ij}]_{n\times n}$. The decision support vector $S^k(D_j) = [s_{ij}^k]_{n\times 1}$ and decision-related vector $R^k(D_j) = [r_{ij}^k]_{n\times 1}$ of x for D_j are Formulated as follows, respectively:

$$s_{ij}^k = \begin{cases} 1, if\ \ [m_i^k]_{n\times 1} \vee G(D_j) = G(D_j) \\ 0, otherwise \end{cases}; \tag{10}$$

$$r_{ij}^k = \begin{cases} 1, if\ \ [m_i^k]_{n\times 1} \wedge G(D_j) \neq 0 \\ 0, otherwise \end{cases}, \tag{11}$$

where "$\vee$" (*or* "$\wedge$") represents the maximum (minimum) value of the corresponding position element in the two vectors.

Definition 3.4. Given an $MDIS = \{S_k|S_k = (U, AT, V_k, F_k), k = 1, 2, \ldots, q\}$, for any $D_j \in U/D$, and $S^k(D_j)$ and $R^k(D_j)$ are the decision support and decision-related vectors of S_k, respectively. The decision support and decision-related matrices under the multi-source information system are defined as $S(D_j) = [S^1(D_j), S^2(D_j), \ldots, S^q(D_j)]_{n\times q}$ and $R(D_j) = [R^1(D_j), R^2(D_j), \ldots, R^q(D_j)]_{n\times q}$, respectively.

Definition 3.5. Given an $MDIS = \{S_k|S_k = (U, AT, V_k, F_k), k = 1, 2, \ldots, q\}$, (α, β) is a pair of thresholds. For any $D_j \in U/D$, the definitions for fixed aggregation vector $MS(D_j)_\alpha = [MS(D_j)^i{}_\alpha]_{n\times 1}$ and possible aggregation vector $MR(D_j)_\beta = [MR(D_j)^i{}_\beta]_{n\times 1}$ of D_j are as follows:

$$MS(D_j)^i{}_\alpha = \begin{cases} 1, if\ \frac{|S(D_j)(x_i)|}{q} \geqslant \alpha \\ 0, otherwise \end{cases}; \tag{12}$$

$$MR(D_j)^i{}_\beta = \begin{cases} 1, if\ \frac{|R(D_j)(x_i)|}{q} > \beta \\ 0, otherwise \end{cases}, \tag{13}$$

where $|S(D_j)(x_i)| = \sum_{k=1}^{q} s_{ij}^k$, $|R(D_j)(x_i)| = \sum_{k=1}^{q} r_{ij}^k$, and $\alpha + \beta = 1$ and $0 < \alpha \leq 1$.

The fixed aggregation operator represents the information fed back from most information sources, and x_i supports decision D_j in a fixed manner; the possible aggregation operator represents the information fed back from various information sources, and x_i may support decision D_j. To better illustrate the static multigranulation fusion strategy in the context of multi-source data, the following examples were constructed to supplement the detailed calculation steps. In addition, Algorithm 1 presents the corresponding static multigranulation fusion algorithm, which describes the steps of the multigranularity method used to obtain information from multisource data.

Example 3.1. A one-way highway involves two observation areas (with multiple observation points), and the observation points are obtained by vehicle attributes (vehicle length, width, height, and weight attributes) and vehicle types (large and small vehicles). Table 1 presents processed observation area data, which is a typical multisource information system, denoted by $MDIS = \{S_i\,|S_i = (U, AT, V_i, F_i), i = 1, 2, 3\}$, where $U = \{x_i|i \in (1, 2, \ldots, 10)\}$, $AT = C \cup D = \{(c_1, c_2, c_3, c_4) \cup d\}$ (vehicle length, width, height, weight, and type), and $U/D = \{D_1, D_2\}$.

Based on the previous definition, the acquisition of two multi-granulation fusion operators from a multisource information system can be divided into three steps. First, decision vector $G(D_j)$ and relation matrix ME^k are calculated. Second, decision support matrix $S(D_j)$ and decision correlation matrix $R(D_j)$ are calculated. Finally, two multi-granulation fusion operators are computed. Fig. 2 shows the detailed process of solving the multi-granulation fusion operator in Example 3.1.

Table 1
An MDIS in Example 3.1.

Object	S_1				S_2				S_3				d
	c_1	c_2	c_3	c_4	c_1	c_2	c_3	c_4	c_1	c_2	c_3	c_4	
x_1	1	2	1	0	1	0	1	0	0	2	1	2	0
x_2	0	2	1	2	2	1	0	1	1	2	0	1	0
x_3	1	0	1	2	1	1	0	2	2	2	1	0	1
x_4	1	2	1	0	1	0	1	0	0	2	1	2	0
x_5	0	2	1	2	0	1	2	1	1	2	0	1	1
x_6	1	0	1	2	1	1	0	2	1	1	2	1	1
x_7	1	2	1	0	0	1	2	1	0	2	1	2	0
x_8	0	2	1	2	2	1	0	1	1	2	0	1	1
x_9	1	0	1	2	1	1	0	2	2	2	1	0	1
x_{10}	1	2	1	0	1	0	1	0	1	1	2	1	0

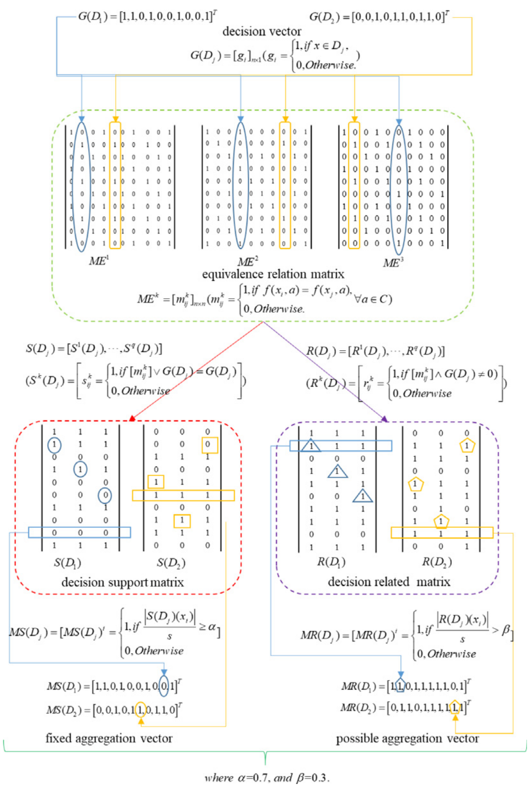

Fig. 2. Multi-granulation fusion operator solution flow chart.

Algorithm 1 is a static algorithm based on matrices for multi-granulation information fusion. Now, we analyze the time complexity of the main steps of the algorithm. According to Definition 3.1, the time complexity of calculating the Boolean vector of decision class D_j in $MDIS$ is $O(|U| \times s)$, particularly steps 1 to 9. According to Definition 3.2, the time complexity of calculating the equivalence relation matrix is $O(|U|^2 \times q)$, and the number of steps is between 10 and 21. In step 23, the time complexity of calculating the decision support matrix $S\left(D_j\right)$ and decision correlation matrix $R\left(D_j\right)$ of each information source through ME^k is $O(|U| \times q \times s)$, as defined by Definitions 3.3 and 3.4. The time complexity of step 24 in computing the fixed aggregation and possible aggregation vectors is $O(|U|)$ according to Definition 3.5. Consequently, the total time complexity of Algorithm 1 is $O(|U|^2 \times q)$.

4. Matrix-based approach for dynamic information fusion in an environment of changing objects

This section presents a matrix-based dynamic information fusion technique that dynamically updates the fixed and possible aggregation operators in response to the addition or removal of objects. To better illustrate dynamic changes for object changes, this section supplements the description using examples. For example, a one-way highway (with a bifurcated intersection) has an observation area at both ends and has multiple observation points in the observation area (the data collected by the corresponding observation points in the two areas are identical). The vehicle attributes (vehicle length, width, height, and weight) collected from the observation points are used to determine the vehicle type passing through a specific area.

Algorithm 1: An algorithm for multi-granulation information fusion that is based on matrices.

Input: $MDIS = \left\{S_1, S_2, \cdots, S_q\right\}$, $U/D = \{D_1, D_2, \cdots, D_s\}$ and threshold (α, β).
Output: The fixed aggregation vector, the possible aggregation vector, the decision support matrix, the relation matrix, and the decision related matrix.

1 **for** $j = 1 : s$ **do**
2 **for** $i = 1 : n$ **do**
3 **if** $x_i \in D_j$ **then**
4 $g_{ij} = 1$;
5 **else**
6 $g_{ij} = 0$;
7 **end**
8 **end**
9 **end**
10 **for** $k = 1 : q$ **do**
11 **for** $j = 1 : n$ **do**
12 **for** $i = 1 : n$ **do**
13 **if** $f(x_i, c) = f(x_j, c)$ **then**
14 $m_{ij}^k = 1$;
15 **else**
16 $m_{ij}^k = 0$;
17 **end**
18 **end**
19 **end**
20 $ME^k = \left[{m^k}_{ij}\right]_{n \times n}$;
21 **end**
22 **for** $each j = 1 : s$ **do**
23 **Compute** the decision support matrix $S\left(D_j\right)$ and decision related matrix $R\left(D_j\right)$ induced by ME^k `// According to Definition 3.3 and 3.4`
24 **Compute** the fixed aggregation vector MS_α and the possible aggregation vector MR_β. `// According to Definition 3.5`
25 **end**

Output: ME^k;$S\left(D_j\right)$; $R\left(D_j\right)$; MS_α; MR_β.

4.1. Matrix-based approach for dynamically updating fixed and possible aggregation operators while adding objects

When numerous objects are added to the multisource data, this subsection primarily focuses on the approach for dynamically updating the fixed and possible aggregation operators based on the matrix. Fig. 3 depicts a schematic of the addition of objects to a multisource information system. For example, in considering how to calculate the fixed and possible aggregation operators only when a vehicle diversion occurs at the intersection in front of a specific area without vehicle influx, we assume that no flow or inflow junctions exist between the specific area and second

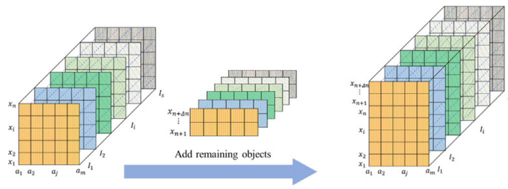

Fig. 3. Schematic diagram of the addition of multi-source information system objects.

observation area, and inflow junction exists only between the first observation area and specific area. Thus, the data collected in the second observation area relative to the data collected in the first observation area constitute a multisource information system with additional objects.

For convenience, we first explain some of the terms used in this definition. We assume that $MDIS^t = \left\{S_k^t | S_k^t = (U^t, AT, V_k^t, F_k^t)\right\}$ and let $MDIS^{t+1} = \left\{S_k^{t+1} | S_k^{t+1} = (U^{t+1}, AT, V_k^{t+1}, F_k^{t+1})\right\}$ be two MDISs, where t and $t+1$ represent the MDIS at time t and $t+1$, respectively. $U^t = \{x_i \,| i \in (1, 2, \ldots, n)\}$, and $U^{t+1} = \{x_i \,| i \in (1, 2, \ldots, n, \ldots, n + \Delta n)\}$ ($U^{t+1} = U \cup \Delta U$, $U = U^t$). Let $MS_\alpha^t = \left[MS(D_1)_\alpha^t, MS(D_2)_\alpha^t, \ldots, MS(D_s)_\alpha^t\right]$ and $MR_\beta^t = \left[MR(D_1)_\beta^t, MR(D_2)_\beta^t, \ldots, MR(D_s)_\beta^t\right]$ denote the fixed and possible aggregation operator matrices at time t, respectively. Let $MS_\alpha^{t+1} = \left[MS(D_1)_\alpha^{t+1}, MS(D_2)_\alpha^{t+1}, \ldots, MS(D_s)_\alpha^{t+1}\right]$ and $MR_\beta^{t+1} = \left[MR(D_1)_\beta^{t+1}, MR(D_2)_\beta^{t+1}, \ldots, MR(D_s)_\beta^{t+1}\right]$ denote the fixed and possible aggregation operator matrices at time $t+1$, respectively.

These terms support the meaning of the partial notation used in the definition, and we now provide the equivalence relation matrix in different states.

Definition 4.1. Let $MDIS^t$ and $MDIS^{t+1}$ be two MDISs; the equivalence relation matrix $ME^{k,t} = \left[m^{k,t}{}_{ij}\right]_{n\times n}$, $ME^{k,t+1}_{(U,\Delta U)} = \left[m^{k,t+1}_{ij,(U,\Delta U)}\right]_{n\times \Delta n}$, $ME^{k,t+1}_{(\Delta U,U)} = \left[m^{k,t+1}_{ij,(\Delta U,U)}\right]_{\Delta n\times n}$, and $ME^{k,t+1}_{(\Delta U,\Delta U)} = \left[m^{k,t+1}_{ij,(\Delta U,\Delta U)}\right]_{\Delta n\times \Delta n}$ are defined as follows:

$$m_{ij}^{k,t} = \begin{cases} 1, iff\, (x_i, c)^t = f\left(x_j, c\right)^t \\ 0, otherwise \end{cases} x_i, x_j \in U; \tag{14}$$

$$m_{ij,(U,\Delta U)}^{k,t+1} = \begin{cases} 1, iff\, (x_i, c)^{t+1} = f\left(x_j, c\right)^{t+1} \\ 0, otherwise \end{cases} x_i \in U, x_j \in \Delta U; \tag{15}$$

$$m_{ij,(\Delta U,U)}^{k,t+1} = \begin{cases} 1, iff\, (x_i, c)^{t+1} = f\left(x_j, c\right)^{t+1} \\ 0, otherwise \end{cases} x_i \in \Delta U, x_j \in U; \tag{16}$$

$$m_{ij,(\Delta U,\Delta U)}^{k,t+1} = \begin{cases} 1, if\, (x_i, c)^{t+1} = f\left(x_j, c\right)^{t+1} \\ 0, otherwise \end{cases} x_i \in \Delta U, x_j \in \Delta U. \tag{17}$$

Definition 4.2. Let $MDIS^t$ and $MDIS^{t+1}$ be two MDISs. For each $D_j \in U/D$, decision vectors $G(D_j)^t = \left[g_{ij}{}^t\right]_{n\times 1}$ and $G(D_j)^{t+1}_{\Delta U} = \left[g_{ij}{}^{t+1}\right]_{\Delta n\times 1}$ are denoted as follows:

$$g_{ij}{}^t = \begin{cases} 1, if\ x_i \in D_j \\ 0, otherwise \end{cases} x_i \in U;\ \ g_{ij}{}^{t+1} = \begin{cases} 1, if x_i \in D_j \\ 0, otherwise \end{cases} x_i \in \Delta U. \tag{18}$$

Proposition 4.1. *Let $MDIS^t$ and $MDIS^{t+1}$ be two MDISs. For each $D_j \in U/D$, the following hold.*

(1) Assuming that $ME^{k,t}$ is the equivalence relation matrix at time t. $ME^{k,t+1}_{(U,\Delta U)}$, $ME^{k,t+1}_{(\Delta U,U)}$, $ME^{k,t+1}_{(\Delta U,\Delta U)}$, and $ME^{k,t+1}$ are equivalence relation matrices at time $t+1$ Then,

$$ME^{k,t+1} = \begin{bmatrix} ME^{k,t} & ME^{k,t+1}_{(U,\Delta U)} \\ ME^{k,t+1}_{(\Delta U,U)} & ME^{k,t+1}_{(\Delta U,\Delta U)} \end{bmatrix}, \tag{19}$$

where equivalence relation matrix $ME^{k,t+1}$ is a symmetric matrix, and $ME^{k,t+1}_{(\Delta U,U)} = (ME^{k,t+1}_{(U,\Delta U)})^T$, 'T' is the transpose of the matrix.

(2) Assuming that $G(D_j)^t$ is the decision vector at time t, $G(D_j)^{t+1}$ and $G(D_j)^{t+1}_{\Delta U}$ are decision vectors at time $t+1$. Then,

$$G(D_j)^{t+1} = \begin{bmatrix} G(D_j)^t \\ G(D_j)^{t+1}_{\Delta U} \end{bmatrix}. \tag{20}$$

Proof. (1) and (2) are easily proved using Definitions 3.1, 3.2, 4.1, and 4.2.

Definition 4.1 provides the equivalence relation matrices between the original object, between the original and newly added objects, and between the newly added object. Simultaneously, Definition 4.2 and Proposition 4.1 clarify that, when calculating the updated equivalence relation matrix, only the last two equivalence relation matrices must be updated, and when updating the decision vector, only the decision vector of the newly added object must be updated. By updating the equivalence relation matrix and decision vector, we can recalculate the decision support and decision-related vectors.

Definition 4.3. Let $MDIS^t$ and $MDIS^{t+1}$ be two MDISs. For any $D_j \in U/D$, $G(D_j)^t$ and $G(D_j)^{t+1}$ are the decision vectors of D_j, respectively. $ME^{k,t+1}$ is the equivalence relation matrix at time $t+1$. Then decision support vector $S^{k,t+1}(D_j) = \left[s_{ij}^{k,t+1}\right]_{(n+\Delta n)\times 1}$ and decision related vector $R^{k,t+1}(D_j) = \left[r_{ij}^{k,t+1}\right]_{(n+\Delta n)\times 1}$ are defined as follows:

$$s_{ij}^{k,t+1} = \begin{cases} 1, if \left[m_i^{k,t+1}\right]_{(n+\Delta n)\times 1} \vee G^{t+1}(D_j) = G^{t+1}(D_j) \\ 0, otherwise \end{cases}; \tag{21}$$

$$r_{ij}^{k,t+1} = \begin{cases} 1, if \left[m_i^{k,t+1}\right]_{(n+\Delta n)\times 1} \wedge G^{t+1}(D_j) \neq \mathbf{0} \\ 0, otherwise \end{cases}; \tag{22}$$

where $\left[m_i^{k,t+1}\right]_{(n+\Delta n)\times 1}$ represents the ith column of $ME_{ij}^{k,t+1}$, and $\mathbf{0}$ is the zero vector.

Definition 4.4. Let $MDIS^t$ and $MDIS^{t+1}$ be two MDISs; $0 < \alpha \leq 1$, $0 \leq \beta < 1$ and $\alpha + \beta = 1$. For any $D_j \in U/D$, the fixed aggregation vector $MS^{t+1}(D_j)_\alpha = \left[MS^{t+1}(D_j)^i{}_\alpha\right]_{(n+\Delta n)\times 1}$ and possible aggregation vector $MR^{t+1}(D_j)_\beta = \left[MR^{t+1}(D_j)^i{}_\beta\right]_{(n+\Delta n)\times 1}$ of D_j are formulated as follows, respectively:

$$MS^{t+1}(D_j)^i{}_\alpha = \begin{cases} 1, if \dfrac{\left|S(D_j)(x_i)^{t+1}\right|}{q} \geqslant \alpha\ \ (i = 1, 2, \ldots, (n+\Delta n)); \\ 0, otherwise \end{cases} \tag{23}$$

$$MR^{t+1}(D_j)^i{}_\beta = \begin{cases} 1, if \dfrac{\left|R(D_j)(x_i)^{t+1}\right|}{q} > \beta,\ \ (i = 1, 2, \ldots, (n+\Delta n)); \\ 0, otherwise \end{cases} \tag{24}$$

where $\left|S(D_j)(x_i)^{t+1}\right| = \sum_{k=1}^q s_{ij}^{k,t+1}$, and $\left|R(D_j)(x_i)^{t+1}\right| = \sum_{k=1}^q r_{ij}^{k,t+1}$. In particular, $\alpha = 0 (\beta = 1)$ implies an optimistic, fixed (possible) aggregation operator; $\alpha = 1\ (\beta = 0)$ indicates a pessimistic, fixed (possible) aggregation operator.

Table 2
The object added to the multi-source information system in Example 4.1.

Object	S_1				S_2				S_3				d
	c_1	c_2	c_3	c_4	c_1	c_2	c_3	c_4	c_1	c_2	c_3	c_4	
x_{11}	1	2	1	0	1	0	1	0	1	1	2	1	0
x_{12}	0	2	1	2	2	1	0	1	1	2	0	1	1
x_{13}	1	2	1	0	1	0	1	0	0	2	1	2	0
x_{14}	0	2	1	2	0	1	2	1	1	2	0	1	0
x_{15}	1	0	1	2	1	1	0	2	1	1	2	1	1
x_{16}	1	0	1	2	1	1	2	1	2	2	1	0	1

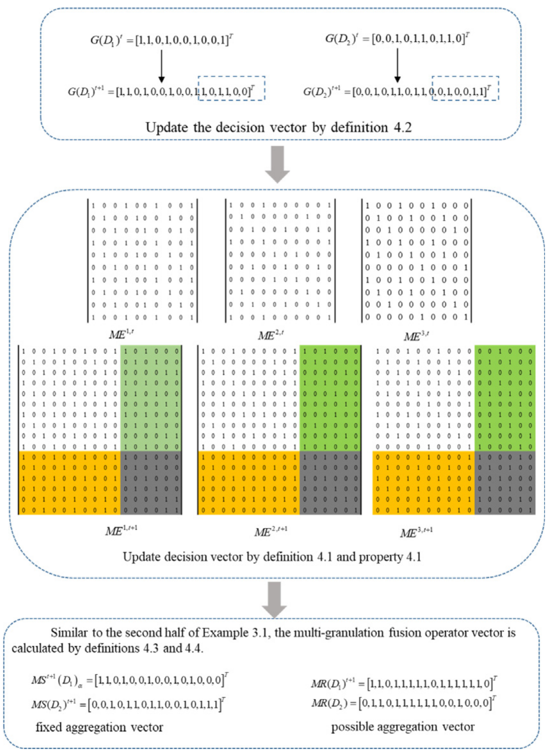

Fig. 4. Flowchart showing the calculation of the multi-granulation fusion operator while adding objects (where $\alpha = 0.70$ and $\beta = 0.30$).

The decision support and decision-related vectors in all information sources are updated according to Definition 4.3, and the two multigranulation fusion operators are then updated according to Definition 4.4. Example 4.1 presents the update process of the two multigranulation fusion operators.

Example 4.1 (*Continue Example 3.1*)**.** Based on Example 3.1, Table 2 presents new vehicle information observed from the second observation area (with the addition of inflow intersections). Tables 1 and 2 constitute a new multisource information system.

The core of the method of dynamically updating multi-granular fusion operators after adding objects is the update of decision vector $G(D_j)^{t+1}$ and equivalence relation matrix $ME^{k,t+1}$. Fig. 4 shows the detailed calculation process of Example 4.1.

Algorithm 2 is a matrix-based multigranulation information fusion algorithm that adds objects to a multisource information system. The temporal complexity of calculating the Boolean vector of decision class D_j when adding objects is $O(|U + \Delta U| \times s)$, particularly for steps 1 to 9. According to Definition 4.1, the temporal complexity of computing the relationship matrix of the new object and that between the new and original objects is $O(|U| \times |\Delta U| \times q)$, and the exact number of steps performed is 10–31. According to Proposition 4.1, the temporal complexity of calculating the relation matrix $ME^{k,t+1}$ at time $t+1$ is $O(q)$, and the actual number of steps performed is 32–34. According to Definition 4.3, the time complexity for computing decision support matrix $S\left(D_j^{t+1}\right)$ and decision-related matrix $R\left(D_j^{t+1}\right)$ from the equivalent relationship matrix $ME^{k,t+1}$ of each source is $O(|U| \times q \times s)$, and the steps are 36. According to Definition 4.4, the temporal complexity of calculating the fixed and possible aggregation vectors is $O(|(U + \Delta U)| \times s)$, and the number of steps is 37. Consequently, the total time complexity of Algorithm 2 is $O(|U| \times |\Delta U| \times q)$.

When adding objects, the static algorithm must recalculate the latest multisource information system; that is, the original object data and newly added objects are merged to calculate an equivalent relation matrix. Based on, when a new object is added, the overall computational time complexity of Algorithm 1 is $O(|U + \Delta U|^2 \times q)$. Thus, the time complexity of the dynamic algorithm is significantly lower than that of the static algorithm.

4.2. Matrix-based approach for dynamically updating fixed and possible aggregation operators while deleting objects

This section primarily focuses on the dynamic updating process of two multisource information fusion operators when objects are deleted from the multisource data. Fig. 5 presents a diagram of the deletion of objects in the multisource data. Similar to Section 4.1, we assumed that no shunting or confluence intersections occur between the specific area and second observation area and only shunting intersections exist between the first observation area and specific area. Thus, the data collected in the second observation area, relative to the data collected in the first observation area, constitute a multisource information system with a reduced number of objects.

Definition 4.5. We assume that $MDIS^t = \left\{S_k^t | S_k^t = (U^t, AT, V_k^t, F_k^t), k = 1, 2, \ldots, q\right\}$ and $MDIS^{t-1} = \left\{S_k^{t-1} | S_k^{t-1} = (U^{t-1}, AT, V_k^{t-1}, F_k^{t-1}), k = 1, 2, \ldots, q\right\}$ represent the original MDIS and the that after deleting the object, respectively. $ME^{k,t-1}$ and $G(D_j)^{t-1}$ represent the equivalence relation matrix and decision vector after deleting the object, respectively.

Proposition 4.2. *Let $MDIS^t$ and $MDIS^{t-1}$ be two MDISs. For any $D_j \in U/D$, the following hold.*

(1) Assuming that $ME^{k,t}$ is the equivalence relation matrix at time t, $ME^{k,t-1}$ is the equivalence relation matrix at time $t-1$; that is, the relation matrix after deleting objects, and $m_i^{k,t}$ represents the relation vector of the deleted object x_i at time t. Then, we have

$$ME_{ij}^{k,t} = \begin{bmatrix} ME_{ij}^{k,t-1} & m_i^{k,t} \\ m_i^{k,t} & 1 \end{bmatrix}. \tag{25}$$

(2) We assume that $G(D_j)^t$ and $G(D_j)^{t-1}$ represent the original decision vector and that after deleting the object, respectively. x_i represents the deleted object, and $G(D_j)^t$ and $G(D_j)^{t-1}$ satisfy

$$G(D_j)^t = \begin{cases} \begin{bmatrix} G(D_j)^{t-1} \\ 0 \end{bmatrix}_{n\times 1}, & if\, x_i \notin D, \\ \begin{bmatrix} G(D_j)^{t-1} \\ 1 \end{bmatrix}_{n\times 1}, & if\, x_i \in D_j. \end{cases} \tag{26}$$

Evidently, Proposition 4.2 can be easily obtained from the elementary transformation properties of the matrix and Definitions 3.1 and 3.2.

According to Proposition 4.2, the equivalence relation matrix after deleting objects can easily be observed to be obtainable by removing the original equivalence relation matrix to reduce the row and column of the object, and the decision vector can be obtained in the corresponding manner. Therefore, the equivalence relation matrix and decision vector are updated.

Algorithm 2: A matrix-based algorithm for multi-granulation information fusion when adding objects.

Input: $MDIS = \{S_1, S_2, \cdots, S_q\}$, $U/D = \{D_1, D_2, \cdots, D_s\}$, the added objects, threshold (α, β) and relation matrix ME^k obtained by algorithm 1.
Output: The fixed aggregation vector, the possible aggregation vector.

for $j = 1 : s$ **do**
  for $i = 1 : n + \Delta n$ **do**
    if $x_i \in D_j$ **then**
      $g_{ij} = 1$;
    else
      $g_{ij} = 0$;
    end
  end
end
for $k = 1 : q$ **do**
  for $j = 1 : \Delta n$ **do**
    for $i = 1 : \Delta n$ **do**
      if $f(x_i, c) = f(x_j, c)$ **then**
        $m_{ij}^k = 1$;
      else
        $m_{ij}^k = 0$;
      end
    end
  end
  $ME^k_{ij,(\Delta U, \Delta U)} = [{m^k}_{ij}]_{\Delta n \times \Delta n}$;
  for $j = 1 : n$ **do**
    for $i = 1 : \Delta n$ **do**
      if $f(x_i, c) = f(x_j, c)$ **then**
        $m_{ij}^k = 1$;
      else
        $m_{ij}^k = 0$;
      end
    end
  end
  $ME^k_{ij,(U, \Delta U)} = [{m^k}_{ij}]_{n \times \Delta n}$
end
for *each* $k = 1 : q$ **do**
  Compute relation matrix $ME^{k,t+1}$;
  // According to Definition 4.1 and Proposition 4.1.
end
for *each* $j = 1 : s$ **do**
  Compute the decision support matrix $S\left(D_j\right)^{t+1}$ and decision related matrix $R\left(D_j\right)^{t+1}$ induced by $ME_{ij}^{k,t+1}$ // According to the definition 4.3 calculated decision support vector and decision related vector, and according to the definition 3.4 calculated decision support matrix and decision related matrix.
  Compute the fixed aggregation vector MS_α^{t+1} and the possible aggregation vector ${MR_\beta}^{t+1}$. // According to Definition 4.43.
end
Output: $ME^{k,t+1}$; ${MS_\alpha}^{t+1}$; ${MR_\beta}^{t+1}$.

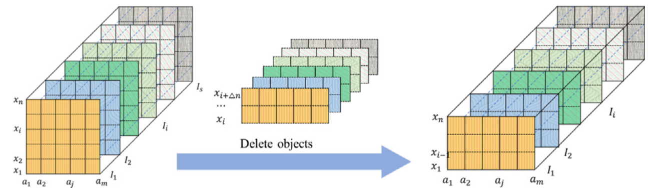

Fig. 5. Schematic diagram of deleting a multi-source information system object.

Definition 4.6. Let $MDIS^t$ and $MDIS^{t-1}$ be two MDISs. $G(D_j)^t$ and $G(D_j)^{t-1}$ are the original decision vector and that after deleting the object, respectively. $ME^{k,t-1}$ denotes the equivalence relation matrix after deleting objects. Decision support vector $S^{k,t-1}(D_j) = \left[s_{ij}^{k,t-1}\right]_{(n-\Delta n)\times 1}$ and decision-related vector $R^{k,t-1}(D_j) = \left[s_{ij}^{k,t-1}\right]_{(n-\Delta n)\times 1}$ are denoted as follows, respectively:

$$s_{ij}^{k,t-1} = \begin{cases} 1, if \left[m_i^{k,t-1}\right]_{(n-\Delta n)\times 1} \vee G^{t-1}(D_j) = G^{t-1}(D_j) \\ 0, otherwise \end{cases}; \tag{27}$$

$$r_{ij}^{k,t-1} = \begin{cases} 1, if \left[m_i^{k,t-1}\right]_{(n-\Delta n)\times 1} \wedge G^{t-1}(D_j) \neq \mathbf{0} \\ 0, otherwise \end{cases}; \tag{28}$$

where $\left[m_i^{k,t-1}\right]_{n\times 1}$ represents the ith column of $ME^{k,t-1}$ after deleting objects, that is, the equivalence class matrix of object x_i after deleting objects.

Definition 4.7. Assume that $MDIS^t$ and $MDIS^{t-1}$ are two MDISs, $0 < \alpha \leq 1$, $0 \leq \beta < 1$ and $\alpha + \beta = 1$. For any $D_j \in U/D$, the fixed aggregation operator vector $MS^{t-1}(D_j)_\alpha = \left[MS^{t-1}(D_j)^i{}_\alpha\right]_{(n-\Delta n)\times 1}$ and possible aggregation operator vector $MR^{t-1}(D_j)_\beta = \left[MR^{t-1}(D_j)^i{}_\beta\right]_{(n-\Delta n)\times 1}$ of D_j are formulated as follows, respectively:

$$MS^{t-1}(D_j)^i{}_\alpha = \begin{cases} 1, if \frac{\left|S(D_j)(x_i)^{t-1}\right|}{q} \geqslant \alpha \\ 0, otherwise \end{cases}, \tag{29}$$

$$MR^{t-1}(D_j)^i{}_\beta = \begin{cases} 1, if \frac{\left|R(D_j)(x_i)^{t-1}\right|}{q} > \beta \\ 0, otherwise \end{cases}, \tag{30}$$

where $\left|S(D_j)(x_i)^{t-1}\right| = \sum_{k=1}^q s_{ij}^{k,t-1}$, and $\left|R(D_j)(x_i)^{t-1}\right| = \sum_{k=1}^q r_{ij}^{k,t-1}$.

Using the updated equivalence relation and Definition 4.6, we obtain the decision support and correlation vectors after deleting the object. By Definition 4.7, the two multigranulation fusion operators are updated after an object is deleted. Example 4.2 presents the updating of two multigranulation fusion operators after objects are deleted.

Example 4.2 (*Continue to Example 3.1*)**.** Based on Example 3.1, Table 3 presents the vehicle information observed from the second observation area (with vehicles flowing out of the diversion junction). In Table 3, the part crossed by the diagonal line is the outflow vehicle at the diversion intersection.

The core of the method of dynamically updating the multigranulation fusion operator after deleting an object is to update decision vector $G(D_j)^{t-1}$ and equivalence relation matrix $ME^{k,t-1}$. Fig. 6 shows the detailed calculation process for Example 4.2.

Algorithm 3 is a matrix-based multigranulation information fusion algorithm for deleting objects from a multisource information system. The temporal complexity of calculating the Boolean

Table 3
Objects deleted from *MDIS*.

Object	S_1				S_2				S_3				d
	c_1	c_2	c_3	c_4	c_1	c_2	c_3	c_4	c_1	c_2	c_3	c_4	
x_1	1	2	1	0	1	0	1	0	0	2	1	2	0
x_2	0	2	1	2	2	1	0	1	1	2	0	1	0
x_3	~~1~~	~~0~~	~~1~~	~~2~~	~~1~~	~~1~~	~~0~~	~~2~~	~~2~~	~~2~~	~~1~~	~~0~~	~~1~~
x_4	~~1~~	~~2~~	~~1~~	~~0~~	~~1~~	~~0~~	~~1~~	~~0~~	~~0~~	~~2~~	~~1~~	~~2~~	~~0~~
x_5	0	2	1	2	0	1	2	1	1	2	0	1	1
x_6	~~1~~	~~0~~	~~1~~	~~2~~	~~1~~	~~1~~	~~0~~	~~2~~	~~1~~	~~1~~	~~2~~	~~1~~	~~1~~
x_7	1	2	1	0	0	1	2	1	0	2	1	2	0
x_8	~~0~~	~~2~~	~~1~~	~~2~~	~~2~~	~~1~~	~~0~~	~~1~~	~~1~~	~~2~~	~~0~~	~~1~~	~~1~~
x_9	1	0	1	2	1	1	0	2	2	2	1	0	1
x_{10}	1	2	1	0	1	0	1	0	1	1	2	1	0

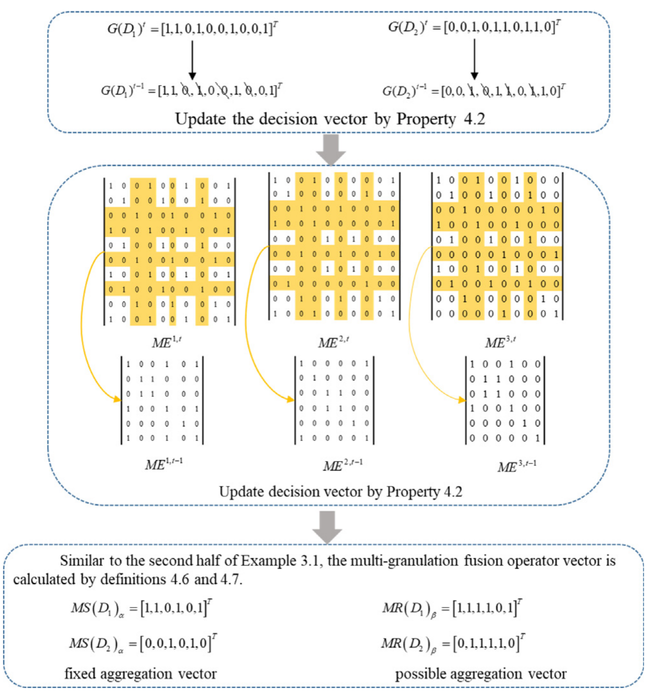

Fig. 6. Calculation flow chart of multi-granulation fusion operator when deleting objects (where $\alpha = 0.7$ and $\beta = 0.3$).

vector of decision class D_j when deleting objects is $O(|U + \Delta U| \times s)$, particularly from steps 1 to 9. According to Property 4.2, the temporal complexity of calculating the relationship matrix $ME^{k,t-1}$ after deleting an object is $O(q)$, and the actual number of steps is 11. According to Definition 4.6, the time complexity of calculating decision support matrix $S\left(D_j{}^{t-1}\right)$ and decision-making matrix $R\left(D_j{}^{t-1}\right)$ from the equivalent relationship matrix $ME^{k,t-1}$ of each information source is $O(|(U - \Delta U)| \times q \times s)$, specifically in 14 steps. According to Definition 4.7, the temporal complexity of calculating the fixed and possible aggregation vectors is $O(|(U - \Delta U)| \times s)$, and the exact number of steps is 15. The total time complexity of Algorithm 3 is $O(|U - \Delta U| \times q \times s)$.

When an object is deleted, the static algorithm must recalculate the equivalence relation matrix of the multisource data. After objects are deleted, the object set of the multi-source data becomes $U - \Delta U$. Evidently, based on Algorithm 1, the overall computational time is $O(|U - \Delta U|^2 \times q)$. The time complexity of the dynamic algorithm is significantly better than that of the static algorithm.

Algorithm 3: A matrix-based algorithm for multi-granulation information fusion when deleting objects.

Input: $MDIS = \{S_1, S_2, \cdots, S_q\}$, $U/D = \{D_1, D_2, \cdots, D_s\}$, the deleted objects, threshold (α, β) and relation matrix ME^k obtained by Algorithm 1.
Output: The fixed aggregation vector, the possible aggregation vector.

```
1  for j = 1 : s do
2      for i = 1 : n − Δn do
3          if x_i ∈ D_j then
4              g_ij = 1;
5          else
6              g_ij = 0;
7          end
8      end
9  end
10 for each k = 1 : q do
11     Compute relation matrix ME^{k,t−1} after deleting
       objects;
       // According to Property 4.2
12 end
13 for eachj = 1 : s do
14     Compute the decision support matrix S(D_j)^{t−1} and
           decision related matrix R(D_j)^{t−1} induced by
           ME^{k,t−1}. // According to the definition 4.6
           calculated decision support vector and
           decision related vector, and according to
           the definition 3.4 calculated decision
           support matrix and decision related
           matrix.
15     Compute the fixed aggregation vector MS_α^{t−1} and the
           possible aggregation vector MR_β^{t−1}. // According
           to Definition 4.7
16 end
   Output: ME^{k^{t−1}}; MS_α^{t−1}; MR_β^{t−1}.
```

5. Matrix-based dynamic information fusion algorithm with varying information sources

In practical applications, the change in information sources is primarily manifested in the increase and decrease of information sources. An increase of information sources means that more information is added to the same object, thereby introducing new content and changes to the overall information of the object. A reduction of information sources means that the information of objects is reduced; that is, changes and differences are introduced to the objects in the information sources. To adapt to changing sources in practical applications, we demonstrate a method based on matrix dynamic fixed and possible aggregation operators. Fig. 7 depicts the information source changes in the multisource information system that this study focused on.

Let $MDIS = \{S_k \, | DIS_k = (U, AT, V, F), k = 1, 2, \ldots, q\}$ be a multisource information system. $MS\left(D_j\right)_\alpha = [f_1{}^\alpha, f_2{}^\alpha, \ldots, f_n{}^\alpha]^T$ and $MR\left(D_j\right)_\beta = \left[r_1{}^\beta, r_2{}^\beta, \ldots, r_n{}^\beta\right]^T$ are Boolean vectors of the fixed and possible aggregation operators, and $S_k(D_j) = \left[S_k{}^1, S_k{}^2, \ldots, S_k{}^n\right]^T$ and $R_k(D_j) = \left[R_k{}^1, R_k{}^2, \ldots, R_k{}^n\right] (k = 1, 2, \ldots, q)$ are the decision support and decision-related vectors of information source k. In addition, $S(D_j) = [S_1, S_2, \ldots, S_q]$ and $R(D_j) = [R_1, R_2, \ldots, R_q]^T$ represent the decision support and decision-making matrices of the multisource information system. $\left|S(D_j)(x_i)\right|$ represents the number of information sources that support x_i, and

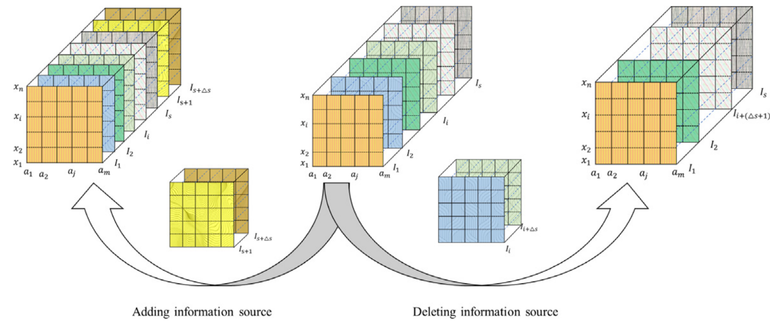

Fig. 7. Diagrammatic representation of the change of information sources inside a *MDIS*.

$\left|R(D_j)(x_i)\right|$ represents the number of information sources that may support x_i, where $\left|*(D_j)(x_i)\right|$ represents the summation of the ith row in the decision support (decision-making) matrix.

5.1. Matrix-based dynamic algorithm for updating fixed and possible aggregation operators when adding information sources

In this subsection, we propose a matrix-based dynamic algorithm for updating the fixed and possible aggregation operators when adding information sources. Adding information sources is always accompanied by an increase in the total information source data, which constitutes a large amount of new useful information. However, neither an increase in information sources nor adding an information source changes the equivalence relation matrix, decision vector, decision support vector, or decision relation vector of the original information sources. Therefore, calculating only the equivalence relation matrix of the new information source and the decision support and decision-related vectors according to Definitions 3.2 and 3.3 is necessary. Then, two multigranulation fusion operators are updated by the decision support and decision-related

Proposition 5.1. *After adding information sources* $\left\{S_{q+1}, S_{q+2}, \ldots, S_{q+p}\right\}$, $MS\left(D_j\right)_{q+p}^{\alpha} = [f_1{}^{\alpha}, f_2{}^{\alpha}, \ldots, f_n{}^{\alpha}]^T$, *and* $MR\left(D_j\right)_{q+p}^{\beta} = \left[r_1{}^{\beta}, r_2{}^{\beta}, \ldots, r_n{}^{\beta}\right]^T$ *denotes the updated Boolean vector.* $S_m(D_j) = \left[S_m{}^1, S_m{}^2, \ldots, S_m{}^n\right]^T$ $(m = 1, 2, \ldots, p)$ *and* $R_m(D_j) = \left[R_m{}^1, R_m{}^2, \ldots, R_m{}^n\right]^T$ *are the decision support and decision-related vectors of the added information source* S_m. $S_{new}(D_j) = [S_1, S_2, \ldots, S_p]^T$, *and* $R_{new}(D_j) = [R_1, R_2, \ldots, R_p]^T$ *denote the decision support and decision-making matrices of the added information sources.*

(1) For each $D_j \in U/D$, *the fixed aggregation operator of* x_i *for* D_j *provides the following results:*

$$f_i(D_j)^{\alpha} = \begin{cases} 1, if \ \frac{|S(x_i)|+|S_{new}(x_i)|}{q+p} \geq \alpha \\ 0, otherwise \end{cases}. \tag{31}$$

(2) For each $D_j \in U/D$, *the possible aggregation operator of* x_i *for* D_j *provides the following results:*

$$r_i(D_j)^{\beta} = \begin{cases} 1, if \ \frac{|R(x_i)|+|R_{new}(x_i)|}{q+p} > \beta \\ 0, otherwise \end{cases}. \tag{32}$$

Proof. (1) Assuming that the original decision support matrix is $S(D_j) = [S_1, S_2, \ldots, S_q]$, and the decision support matrix of the added information source is $S_{new}(D_j) = [S_1, S_2, \ldots, S_p]$, we note the decision support matrix with the new information source is evidently $S_{final}(D_j) = [S_1, S_2, \ldots, S_q, S_{q+1}, S_{q+2}, \ldots, S_{q+p}]$ according to Definition 3.4. Similarly, assuming that the original decision support matrix is $S(D_j) = [S_1, S_2, \ldots, S_q]$, and the decision support matrix of the added information source is $S_{new}(D_j) = [S_1, S_2, \ldots, S_p]$, we note that the decision support matrix with the new information source is $S_{final}(D_j) = [S_1, S_2, \ldots, S_q, S_{q+1}, S_{q+2}, \ldots, S_{q+p}]$ according to Definition 3.4. Thus, based on Definition 3.5, we easily observe that

$$MS_{final}(D_j)^i{}_{\alpha} = \begin{cases} 1, if \ \frac{|S_{final}(D_j)(x_i)|}{q+p} \geqslant \alpha \\ 0, otherwise \end{cases}.$$

Therefore, when $MS_{final}(D_j)^i{}_{\alpha} = 1$, $\frac{|S_{final}(D_j)(x_i)|}{q+p} \geqslant \alpha$. Because

$$\sum_{k=1}^{q+p} S_k(D_j)(x_i) = \sum_{k=1}^{q} S_k(D_j)(x_i) + \sum_{k=1}^{p} S_k(D_j)(x_i) = \left|S(D_j)(x_i)\right| + \left|S_{new}(D_j)(x_i)\right|,$$

$MS_{final}(D_j)^i{}_{\alpha} = f_i(D_j)^{\alpha}$. When $MS_{final}(D_j)^i{}_{\alpha} = 0$, $MS_{final}(D_j)^i{}_{\alpha} = f_i(D_j)^{\alpha}$ holds.

(2) Assuming that the original decision related matrix is $R(D_j) = [R_1, R_2, \ldots, R_q]$, and the decision related matrix of the added information source is $R_{new}(D_j) = [R_1, R_2, \ldots, R_p]$, Definition 3.4 provides that the decision related matrix with the new information source is $R_{final}(D_j) = [R_1, R_2, \ldots, R_q, R_{q+1}, R_{q+2}, \ldots, R_{q+p}]$. Thus, Definition 3.5 provides that

$$MR_{final}(D_j)^i{}_{\beta} = \begin{cases} 1, if \ \frac{|R_{final}(D_j)(x_i)|}{q+p} > \beta \\ 0, otherwise \end{cases}.$$

Therefore, when $MR_{final}(D_j)^i{}_{\beta} = 1$, $\frac{|R_{final}(D_j)(x_i)|}{q+p} > \beta$. Because

$$\sum_{k=1}^{q+p} R_k(D_j)(x_i) = \sum_{k=1}^{q} R_k(D_j)(x_i) + \sum_{k=1}^{p} R_k(D_j)(x_i) = \left|R(D_j)(x_i)\right| + \left|R_{new}(D_j)(x_i)\right|,$$

$MR_{final}(D_j)^i{}_{\beta} = r_i(D_j)^{\beta}$. When $MR_{final}(D_j)^i{}_{\beta} = 0$, $MR_{final}(D_j)^i{}_{\beta} = r_i(D_j)^{\beta}$ holds.

As shown in Property 5.1, the two multigranulation fusion operators of the multisource information system after adding the information source have a certain quantitative relationship with the original multigranulation fusion operators and those of the added information source. Therefore, we must only calculate the corresponding decision support and decision-related vectors by adding the new information source and combine those the original multi-source information system to update the two multigranulation fusion operators. Example 5.1 presents the calculation process.

Example 5.1 (*Continue to Example* 3.1)**.** Based on Example 3.1, two observation points are added in the observation area to obtain the information of passing vehicles caused by the analysis requirements. Table 4 presents the data obtained from the added observation points.

The core of the method of dynamically updating multigranulation fusion operators after adding information sources is to calculate two fusion operators using the decision support and decision-related matrices of the new information sources combined with the original decision support and decision-related matrices. Fig. 8 illustrates the calculation process for adding information sources.

Algorithm 4: A matrix-based multi-granulation information fusion algorithm when adding information sources

Input: (1)Original multi-source information system $MDIS = \{S_1, S_2, \cdots, S_q\}$, added information system$MDIS = \{S_1, S_2, \cdots, S_p\}$, $U/D = \{D_1, D_2, \cdots, D_s\}$(2) threshold (α, β), the decision support matrix $S(D_j)$ and decision related matrix $R(D_j)$.

Output: The fixed aggregation vector, the possible aggregation vector.

```
1  for j = 1 : s do
2      for i = 1 : n do
3          if x_i ∈ D_j then
4              g_ij = 1;
5          else
6              g_ij = 0;
7          end
8      end
9  end
10 for k = 1 : p do
11     for j = 1 : n do
12         for i = 1 : n do
13             if f(x_i, c) = f(x_j, c) then
14                 m^k_ij = 1;
15             else
16                 m^k_ij = 0;
17             end
18         end
19     end
20     ME^k = [m^k_ij]_{n×n};
21 end
22 for each j = 1 : s do
23     for k = 1 : p do
24         Compute the decision support matrix S(D_j) and
               decision related matrix R(D_j) induced by ME^k
               ;// According to Definition 3.3 and 3.4
25     end
26     Compute the fixed aggregation vector MS_α and the
           possible aggregation vector MR_β. // According
           to Proposition 5.1
27 end
```

Output: ME^k; MS_α; MR_β.

Algorithm 4 is a multi-granulation information fusion algorithm based on matrix calculations for adding information sources to $MDIS$. The temporal complexity of computing the Boolean vector of decision class D_j when adding information sources to a multisource information system is $O(|U| \times s)$, particular for steps 1–9. According to Definition 3.2, the temporal complexity of constructing the equivalent relation matrix MME^k for each additional information source is $O(|U|^2 \times p)$, and the number of steps is between 10 and 21. According to Definitions 3.3 and 3.4, the temporal complexity of calculating decision support matrix $S(D_j)$ and decision-related matrix $R(D_j)$ for ME^k the additional information sources is $O(|U| \times p \times s)$, and the step count is 23–25. Step 26 calculates the fixed and possible aggregation vectors according to Proposition 5.1, and its time complexity is $O(|U| \times s)$. Therefore, the total time complexity of Algorithm 4 is $O(|U|^2 \times p)$.

Table 4
New information source data in Example 5.1.

Object	S_4				S_5			
	c_1	c_2	c_3	c_4	c_1	c_2	c_3	c_4
x_1	1	2	1	0	1	0	1	0
x_2	1	2	1	0	1	0	1	0
x_3	1	0	1	2	1	1	2	1
x_4	0	2	1	2	0	1	2	1
x_5	0	2	1	2	2	1	0	1
x_6	1	0	1	2	1	1	0	2
x_7	1	1	2	1	0	2	1	2
x_8	1	2	0	1	1	1	2	1
x_9	2	2	1	0	2	1	0	1
x_{10}	1	2	0	1	0	2	1	2

When adding an information source, the dynamic algorithm must only calculate the decision support matrix and decision correlation vector of the additional information source. However, the static algorithm must recalculate the decision support matrix and decision correlation vector of all information sources. Note that the overall number of information sources of multisource data becomes $q+p$, and the time complexity of its static algorithm is $O(|U|^2 \times (q+p))$. Evidently, for $q+p > p$, the time complexity of the dynamic algorithm is lower than that of the static algorithm.

5.2. Matrix-based dynamic algorithm for updating fixed aggregation and possible aggregation operators when deleting information sources

In a multi-source information system, updating the multigranulation fusion operator in the original multi-source information system is necessary because of the failure of some information sources in some states. When the information source is deleted, less information is available as the number of information sources decreases. The original decision support and decision-related matrices exhibit corresponding changes owing to the deletion of information sources. Therefore, the decision support and decision-related vectors must be recalculated according to Definitions 3.2 and 3.3 to delete part of the information sources. To delete sources from the multisource data, we developed methods to dynamically update the two multigranulation fusion operators.

Proposition 5.2. *After deleting the information sources* $\{S_1, S_2, \ldots, S_p\}$, $MS(D_j)_{q-p}^{\alpha} = [f_1{}^{\alpha}, f_2{}^{\alpha}, \ldots, f_n{}^{\alpha}]^T$, *and* $MR(D_j)_{q-p}^{\beta} = [r_1{}^{\beta}, r_2{}^{\beta}, \ldots, r_n{}^{\beta}]^T$ *denotes the updated Boolean vector.* $S_k(D_j) = [S_k{}^1, S_k{}^2, \ldots, S_k{}^n]^T$ $(k = 1, 2, \ldots, q-p)$ *and* $R_k(D_j) = [R_k{}^1, R_k{}^2, \ldots, R_k{}^n]^T$ $(k = 1, 2, \ldots, q-p)$ *denote the decision support and decision-related vectors, respectively.* $S_{deleting}(D_j) = [S_1, S_2, \ldots, S_p]^T$ *and* $R_{deleting}(D_j) = [R_1, R_2, \ldots, R_p]$ *denote the decision support and decision-related matrices required to delete the information sources.* $|S_{deleting}(D_j)(x_i)|$ *represents the number of information sources supporting* x_i, *and* $|R_{deleting}(D_j)(x_i)|$ *represents the number of information sources that may support* x_i.

(1) For each $D_j \in U/D$, *the fixed aggregation operator of* x_i *for* D_j *provides the following results:*

$$f_i(D_j)^{\alpha} = \begin{cases} 1, \text{ if } \frac{|S(x_i)| - |S_{deleting}(x_i)|}{q-p} \geq \alpha \\ 0, \text{ otherwise} \end{cases} . \quad (33)$$

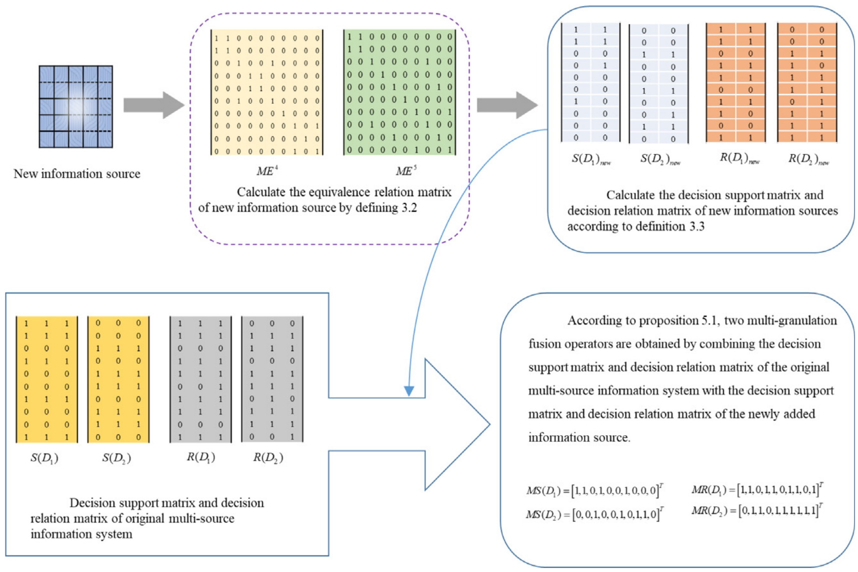

Fig. 8. Calculation flow chart of multi-granulation fusion operator when adding information source (where $\alpha = 0.7$ and $\beta = 0.30$).

(2)For each $D_j \in U/D(j = 1, 2, \ldots, s)$, the possible aggregation operator of x_i for D_j provides the following results:

$$r_i(D_j)^{\beta} = \begin{cases} 1, \text{ if } \frac{|R(x_i)| - |R_{deleting}(x_i)|}{q-p} > \beta \\ 0, \text{ otherwise} \end{cases}. \tag{34}$$

Proof. (1) Assuming that the original support matrix is $S(D_j) = [S_1, S_2, \ldots, S_q]$, and the decision support matrix of the deleted information source is $S_{deleting}(D_j) = [S_1, S_2, \ldots, S_p]$, Definition 3.4 provides that the decision support matrix with the new information source is $S_{final}(D_j) = [S_1, S_2, \ldots, S_{q-p}]$. Based on Definition 3.5,

$$MS_{final}(D_j)^i_{\alpha} = \begin{cases} 1, \text{ if } \frac{|S_{final}(D_j)(x_i)|}{q-p} \geqslant \alpha \\ 0, \text{ otherwise} \end{cases}.$$

Therefore, when $MS_{final}(D_j)^i_{\alpha} = 1$, $\frac{|S_{final}(D_j)(x_i)|}{q-p} \geqslant \alpha$. Because

$$\sum_{k=1}^{q-p} S_k(D_j)(x_i) = \sum_{k=1}^{q} S_k(D_j)(x_i) - \sum_{k=1}^{q} S_k(D_j)(x_i) = \left|S(D_j)(x_i)\right| - \left|S_{deleting}(D_j)(x_i)\right|,$$

$MS_{final}(D_j)^i_{\alpha} = f_i(D_j)^{\alpha}$. When $MS_{final}(D_j)^i_{\alpha} = 0$, $MS_{final}(D_j)^i_{\alpha} = f_i(D_j)^{\alpha}$ holds.

(2) Assuming that the original related decision is $R(D_j) = [R_1, R_2, \ldots, R_q]$, and that of the deleted information source is $R_{deleting}(D_j) = [R_1, R_2, \ldots, R_p]$, Definition 3.4 provides that the decision related matrix with the new information source is $R_{final}(D_j) = [R_1, R_2, \ldots, R_{q-p}]$. Then, based on Definition 3.5,

$$MR_{final}(D_j)^i_{\beta} = \begin{cases} 1, \text{ if } \frac{|R_{final}(D_j)(x_i)|}{q-p} > \beta \\ 0, \text{ otherwise} \end{cases}.$$

Therefore, when $MR_{final}(D_j)^i_{\beta} = 1$, $\frac{|R_{final}(D_j)(x_i)|}{q-p} > \beta$. Because

$$\sum_{k=1}^{q-p} R_k(D_j)(x_i) = \sum_{k=1}^{q} R_k(D_j)(x_i) - \sum_{k=1}^{p} R_k(D_j)(x_i) = \left|R(D_j)(x_i)\right| - \left|R_{deleting}(D_j)(x_i)\right|,$$

$MR_{final}(D_j)^i_{\beta} = r_i(D_j)^{\beta}$. When $MR_{final}(D_j)^i_{\beta} = 0$, $MR_{final}(D_j)^i_{\beta} = r_i(D_j)^{\beta}$ holds.

As shown by Property 5.2, a certain quantitative relationship exists between the two multigranulation fusion operators of the multisource information system after the source is deleted and the original multigranulation fusion operators. Therefore, we must only delete the corresponding decision support and decision-related vectors of the information source and use the original decision support and decision-related matrices to update the two multigranulation fusion operators. Example 5.2 describes the calculation process.

Example 5.2 (*Continue to Example 3.1*)**.** Based on Example 3.1, owing to the equipment failure of one observation point in the observation area, the previously calculated multigranularity fusion operator must be recalculated. Information source S_2 in Table 1 was damaged by the equipment at the observation point.

The core of the method of dynamically updating multi-granulation fusion operators after deleting information sources is to calculate the decision support and correlation matrices of the information sources deleted. Then, we use Property 5.2 to calculate two multi-granulation fusion operators. Fig. 9 illustrates the detailed calculation process for adding information sources.

Algorithm 5 is a multigranulation information fusion algorithm based on matrix calculations for deleting information sources in *MDIS*. The temporal complexity of computing the Boolean vector of decision class D_j in a multi-source information system is $O(|U| \times s)$, and the particular steps are 1–9. According to Definition 3.2, the temporal complexity of calculating the equivalent relation matrix ME^k for each deleted information source is $O(|U|^2 \times p)$, and the steps are 10–21. According to Definitions 3.3 and 3.4, the temporal complexity of calculating decision support matrix $S\left(D_j\right)$ and decision-related matrix $R\left(D_j\right)$ for each deleted source of information is $O(|U| \times p \times s)$, and the steps are 23–25. Step 26 calculates the fixed and possible aggregation vectors according to Proposition 5.2, and its time complexity is $O(|U| \times s)$. Therefore, the total time complexity of Algorithm 5 is $O(|U|^2 \times p)$.

When deleting an information source, the dynamic algorithm must only calculate the decision support matrix and correlation vector of the deleted information source. However, the static algorithm must calculate the decision support matrix and correlation vector of the remaining information sources. The time complexity of the static algorithm is $O(|U|^2 \times q - p)$. Evidently, when $p \leq \frac{q}{2}$, the time complexity of the dynamic algorithm is

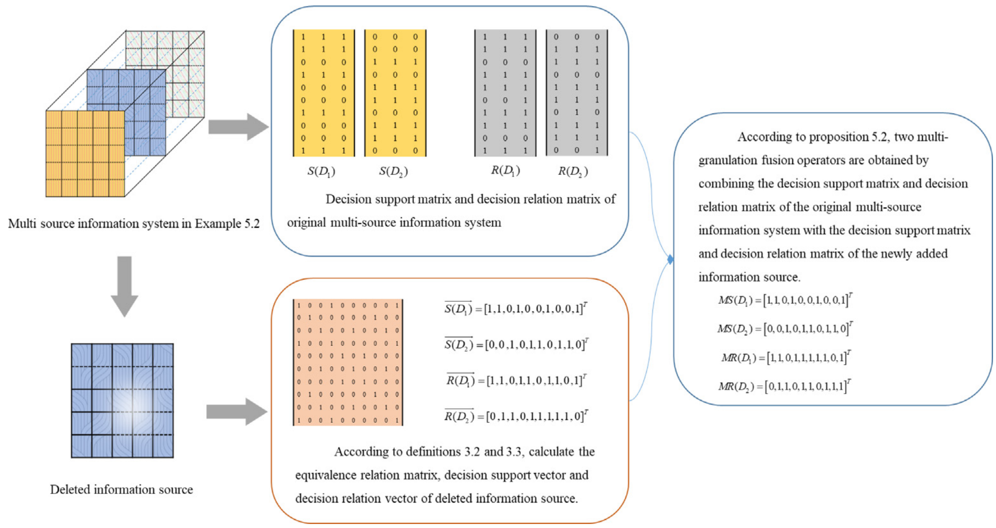

Fig. 9. Calculation flow chart of multi-granulation fusion operator when deleting information source (where $\alpha = 0.7$ and $\beta = 0.30$).

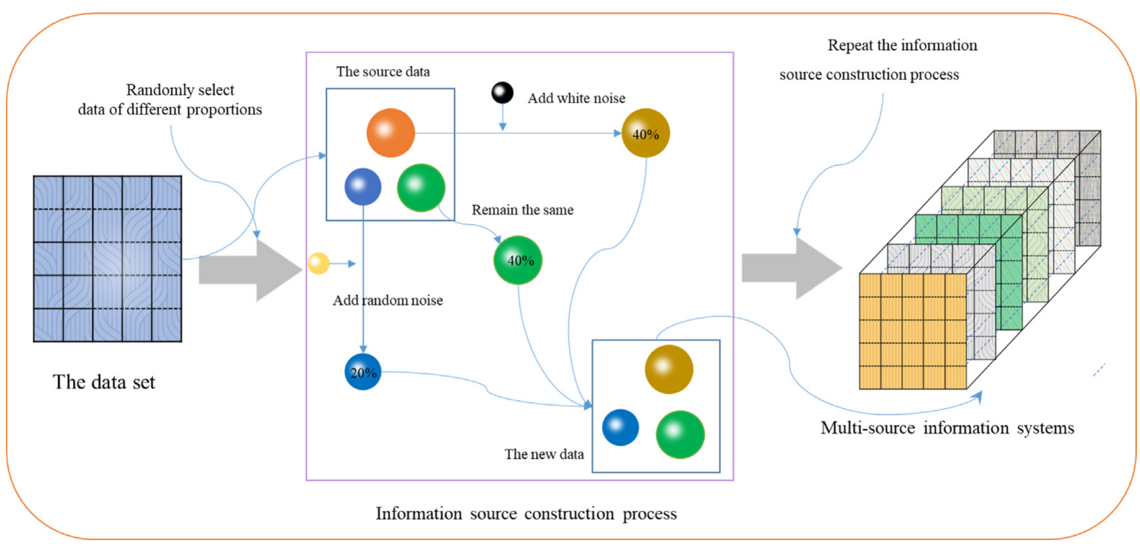

Fig. 10. The construction process of multi-source information system.

better than that of the static algorithm; that is, when the number of sources reduced is less than half of all sources, the computation time of the dynamic algorithm is better than that of the static algorithm.

6. Experimental analysis

In the fourth and fifth sections of this paper, based on the multi-granulation information fusion method, we propose a matrix-based dynamic multi-source information fusion method for adding objects, deleting objects, adding information sources, and deleting information sources. This section presents the tests conducted on the performance and viability of the matrix-based dynamic information fusion algorithm. We used the following computer configuration environment for the experiments. The CPU consisted of an $Intel(R)Core(TM)i5-8250UCPU@1.60GHz$ 1.80 GHz with 4.0GB of RAM. The operating system was $Windows \times 64$. All relevant experiments were conducted using Python 3.5. Ten open datasets were used in the experiments to verify the computational complexity and validity of the dynamic algorithms. All datasets were obtained from the machine learning library UCI (https://archive.ics.uci.edu/ml/index.php); Table 5 lists the details of the datasets. Table 5 includes the name of the dataset, sample size (i.e., the number of objects) in the dataset, number of attributes in the dataset, and total number of categories of data in the dataset (the data can be divided into several categories).

Obtaining a multi-source decision information system directly from a machine learning database is not easy when conducting experiments. To solve this problem, Yang et al. constructed multisource information system data by adding white and random noise to the dataset [48]. The construction method is as follows. First, generate k normally distributed random numbers $(n_1, n_2, \ldots, n_k)$ and k uniformly distributed random numbers $(u_1, u_2, \ldots, u_k)$. Second, add random and white noise as follows:

$$S_i(x,a) = \begin{cases} s(x,a) + n_i, \, if\, 0 \le |n_i| \le 1 \\ s(x,a), \, otherwise \end{cases}; \tag{35}$$

$$S_i(x,a) = \begin{cases} s(x,a) + u_i, \, if\, 0 \le |u_i| \le 1 \\ s(x,a), \, otherwise \end{cases}, \tag{36}$$

where $s(x,a)$ represents the value in the original dataset and $S_i(x,a)$ the data after adding noise. In this study, we chose to add random noise to 40% of the original dataset, white noise to 20% of the dataset, and retain the remaining 40% to build a multi-source

Algorithm 5: A matrix-based multi-granulation information fusion algorithm when deleting information sources

Input: (1)Original multi-source information system $MDIS = \{S_1, S_2, \cdots, S_q\}$, deleted information system$MDIS = \{S_1, S_2, \cdots, S_p\}$, $U/D = \{D_1, D_2, \cdots, D_s\}$;
(2) threshold (α, β), the decision support matrix $S(D_j)$ and decision related matrix $R(D_j)$.
Output: The fixed aggregation vector, the possible aggregation vector.

for $j = 1 : s$ **do**
  for $i = 1 : n$ **do**
    if $x_i \in D_j$ **then**
      $g_{ij} = 1$;
    else
      $g_{ij} = 0$;
    end
  end
end
for $k = 1 : p$ **do**
  for $j = 1 : n$ **do**
    for $i = 1 : n$ **do**
      if $f(x_i, c) = f(x_j, c)$ **then**
        $m^k_{ij} = 1$;
      else
        $m^k_{ij} = 0$;
      end
    end
  end
  $ME^k = [m^k{}_{ij}]_{n\times n}$;
end
for *each* $j = 1 : s$ **do**
  for $k = 1 : p$ **do**
    Compute the decision support matrix $S(D_j)$ and decision related matrix $R(D_j)$ induced by ME^k ;// According to Definition 3.3 and 3.4
  end
  Compute the fixed aggregation vector MS_α and the possible aggregation vector MR_β. // According to Proposition 5.2
end
Output: ME^k; MS_α; MR_β.

information system. Fig. 10 presents the process of building the multisource information system discussed in this section.

In this study, we constructed 10 sources for experiments on adding and deleting objects and 20 sources for experiments on adding and deleting sources. In addition, this study aimed to investigate the dynamic multigranulation information fusion technique in the context of an equivalence relation. Therefore, to satisfy the equivalence conditions, this study discretized the multisource information system generated by the aforementioned method. The discretization method primarily adopts an equal frequency discretization. If the object under attribute a is evenly divided into p parts $X_1(a), X_1(a), \ldots, X_p(a)$, $U = X_1(a) \cup X_1(a) \cup \cdots \cup X_p(a)$ and $\frac{|X_1(a)|}{|U|} = \frac{|X_2(a)|}{|U|} = \cdots = \frac{|X_p(a)|}{|U|}$; the mathematical expression is as follows: $S_i(x, a) = p, ifx \in X_p(a)$. In addition, we used $\alpha = 0.7$ and $\beta = 0.3$ in the experiment comparing the dynamic and static algorithms.

In this section, we describe five groups of experiments conducted for four dynamic situations. Fig. 11 and Tables 6C11 present the results. In Fig. 11 $a-j$, the x-axis indicates the proportion of data changes in the multisource information system, the y-axis indicates the manner in which the data changes (adding objects, deleting objects, adding sources, and deleting sources) and the corresponding static algorithms, and the z-axis indicates the algorithm running time. In Figs. 11 $k-t$, the x-axis indicates the data change ratio of the five experiments, and the y-axis indicates the data change operation performed, which indicates the speedup ratio of the dynamic and static algorithms in the five experiments. The speedup ratio is expressed as $T_{static}/T_{dynamic}$, where T_{static} is the running time of the static algorithm, and $T_{dynamic}$ is the running time of the dynamic algorithm.

6.1. Matrix-based static and dynamic multigranulation fusion algorithms comparison when the object changes

This section discusses the five groups of comparative experiments conducted on adding and deleting objects. In the experiment on object addition, we randomly divided the objects in the 10 constructed information sources into two parts: the original multisource information system and the objects added to the multisource information system in different proportions (accounting for 10%, 20%, 30%, 40%, and 50% of all dataset objects). Simultaneously, the static algorithm followed the algorithm described in Section 3 to calculate the multisource information system formed by adding objects of different proportions. In the object deletion experiment, we randomly deleted objects of different proportions from the constructed multisource information system, and the objects deleted from each information source were the same. The static algorithm of this experiment group followed the algorithm described in Section 3 to calculate the multisource information system after deleting an object. The following two sections present the results of the two experiment groups.

(1) When adding objects, a comparison of the static and dynamic algorithms. Fig. 11 and Table 6 present the results. As shown in Figs. 11 $a-j$ and Table 6, both the dynamic update strategy and static algorithm require more time for the calculation as the number of new objects increases. The graph shows that the dynamic method has a substantially shorter execution time than the static method. In addition, as shown in Figs. 11 $j-t$, as the proportion of added objects gradually increases, the overall speedup ratio of the algorithm exhibits a downward trend. The speedup ratio for the entire method remains greater than 1; therefore, the matrix-based dynamic approach remains faster than the static algorithm.

(2) When deleting objects, a comparison of static and dynamic algorithms. Fig. 11 shows the results as the percentage of deleted objects increases. Figs. 11 $a-j$ and Table 7 indicate that the dynamic updating approach based on a matrix and the static algorithm both run faster as the percentage of removed objects increases. The dynamic methods update the fixed and possible aggregation operators substantially quicker than the static algorithms when the number of objects deleted from the original data steadily increases. Figs. 11 $j-t$ show that the speedup ratio of the dynamic and static algorithms is greater than 1.

In addition, to clarify the benefits of the dynamic algorithm, Table 8 provides the speedup ratio of the running times of the static and dynamic algorithms for various ratios. As shown in Table 8, the average ratio of the speedup between the dynamic and static algorithms in the experiments with added objects ranges from 1.93 to 5.52. Similarly, the average speedup ranges from 4.45 to 25.76 in the experiments with deleted objects. Therefore, we can conclude from Fig. 11 and Table 5 that the computational efficiency of the dynamic algorithm is superior to that of the static approach, which is compatible with the theoretical analysis presented in Section 4.

Table 5
Detailed information of selected datasets.

No.	Datasets	Abbreviation	Samples	Attributes	Decision classes
1	Yach Hydrodynamics	YH	308	6	3
2	Concrete	Concrete	1029	8	10
3	Airfoil Self Noise	AF	1504	5	10
4	Winequality-Red	Red	1599	11	6
5	Winequality-White	White	4098	11	7
6	Abalone	Abalone	4177	7	3
7	Wilt	Wilt	4839	5	2
8	Combined Cycle Power Plant	CP	9569	4	10
9	Crowdsourced Mapping	CM	10845	28	6
10	Dry Bean	DB	13611	16	7

Table 6
Running time of each group of experiments when adding objects.

Datasets	10%		20%		30%		40%		50%	
	Static	Dynamic	Static	Dynamic	Static	Dynamic	Static	Dynamic	Static	Dynamic
Yach Hydrodynamics	2.17	0.59	2.44	1.22	2.78	1.58	3.16	2.16	3.55	2.72
Winequality-White	77.31	20.31	88.36	44.98	101.50	61.84	155.47	84.22	177.39	111.41
Winequality-Red	18.67	3.59	21.80	6.97	25.72	9.88	29.67	13.42	33.30	17.00
Wilt	64.67	14.72	80.72	27.78	91.25	41.91	162.58	53.64	206.28	86.81
Dry Bean	525.30	110.36	735.11	209.70	950.86	449.20	1164.83	629.58	1802.11	664.78
Crowdsourced Mapping	467.69	104.14	654.89	198.03	791.02	293.44	1113.59	420.52	1390.78	614.52
Concrete	7.78	2.34	8.97	4.30	10.14	6.13	11.64	8.34	13.27	10.63
Combined Cycle Power Plant	399.72	61.11	599.38	107.03	809.31	158.11	1172.27	234.59	1489.38	278.47
Airfoil Self Noise	11.23	3.50	13.47	6.64	16.45	9.44	18.56	13.45	21.14	16.55
Abalone	52.64	12.77	64.83	25.41	78.52	35.31	90.72	59.75	139.09	78.73

Table 7
Running time of each group of experiments when deleting objects.

Datasets	10%		20%		30%		40%		50%	
	Static	Dynamic	Static	Dynamic	Static	Dynamic	Static	Dynamic	Static	Dynamic
Yach Hydrodynamics	2.56	0.08	2.31	0.08	2.03	0.08	1.72	0.08	1.44	0.08
Winequality-White	106.92	12.09	97.34	9.70	83.50	7.33	69.92	5.33	53.14	4.05
Winequality-Red	29.16	1.23	22.00	1.13	19.42	0.72	16.23	0.56	8.14	0.44
Wilt	86.08	4.28	81.73	3.72	64.59	2.75	55.89	2.16	42.92	1.58
Dry Bean	891.75	145.78	796.59	138.48	547.27	102.56	438.42	73.59	240.19	49.31
Crowdsourced Mapping	878.58	81.94	696.38	61.39	568.38	34.16	401.17	33.78	298.38	17.45
Concrete	9.41	0.73	8.61	0.61	7.17	0.48	6.09	0.34	4.94	0.23
Combined Cycle Power Plant	274.61	71.97	240.53	57.80	202.44	40.75	160.73	37.83	110.13	21.75
Airfoil Self Noise	18.20	2.50	16.86	1.88	13.44	1.44	11.22	1.03	8.95	0.72
Abalone	83.31	5.17	68.75	3.92	59.19	3.02	51.67	2.42	35.97	1.64

Table 8
Speedup ratio for static and dynamic algorithms when objects change.

Variation ratio	Adding objects						Deleting objects					
	10%	20%	30%	40%	50%	Average	10%	20%	30%	40%	50%	Average
Yach Hydrodynamics	3.66	2.00	1.76	1.46	1.30	2.04	32.80	29.60	26.00	22.00	18.40	25.76
Winequality-White	3.81	1.96	1.64	1.85	1.59	2.17	8.84	10.03	11.39	13.12	13.13	11.30
Winequality-Red	5.20	3.13	2.60	2.21	1.96	3.02	23.62	19.56	27.02	28.86	18.61	23.53
Wilt	4.39	2.91	2.18	3.03	2.38	2.98	20.11	21.98	23.49	25.92	27.20	23.74
Dry Bean	4.76	3.51	2.12	1.85	2.71	2.99	6.12	5.75	5.34	5.96	4.87	5.61
Crowdsourced Mapping	4.49	3.31	2.70	2.65	2.26	3.08	10.72	11.34	16.64	11.88	17.10	13.54
Concrete	3.32	2.09	1.66	1.40	1.25	1.94	12.81	14.13	14.81	17.73	21.07	16.11
Combined Cycle Power Plant	6.54	5.60	5.12	5.00	5.35	5.52	3.82	4.16	4.97	4.25	5.06	4.45
Airfoil Self Noise	3.21	2.03	1.74	1.38	1.28	1.93	7.28	8.99	9.35	10.88	12.46	9.79
Abalone	4.12	2.55	2.22	1.52	1.77	2.44	16.11	17.53	19.63	21.34	21.92	19.30

6.2. Matrix-based static and dynamic multigranulation fusion algorithms comparison when the information source changes

This section discusses the five groups of comparative experiments conducted on adding and deleting information sources. For the information source addition experiment, we constructed 20 information sources, of which 10 sources constituted the original multi-source information system, and the other 10 sources were added to the multisource information system according to the number of information sources selected each time (the number of information sources added was 2, 4, 6, 8, and 10). In addition, the static algorithm followed the algorithm presented in Section 3 to calculate the multisource information system after adding information sources, and its running time was compared with that of the dynamic algorithm. In the information source deletion experiment, we used the 20 information sources constructed as the original multisource information system and deleted different numbers of information sources (the numbers of deleted information sources were 2, 4, 6, 8, and 10) each experiment. The static algorithm followed that presented in Section 3 to calculate the multi-source information system after deleting the information source, and we compare its running time with that of the dynamic algorithm. The following two sections present the results of the two experiment groups.

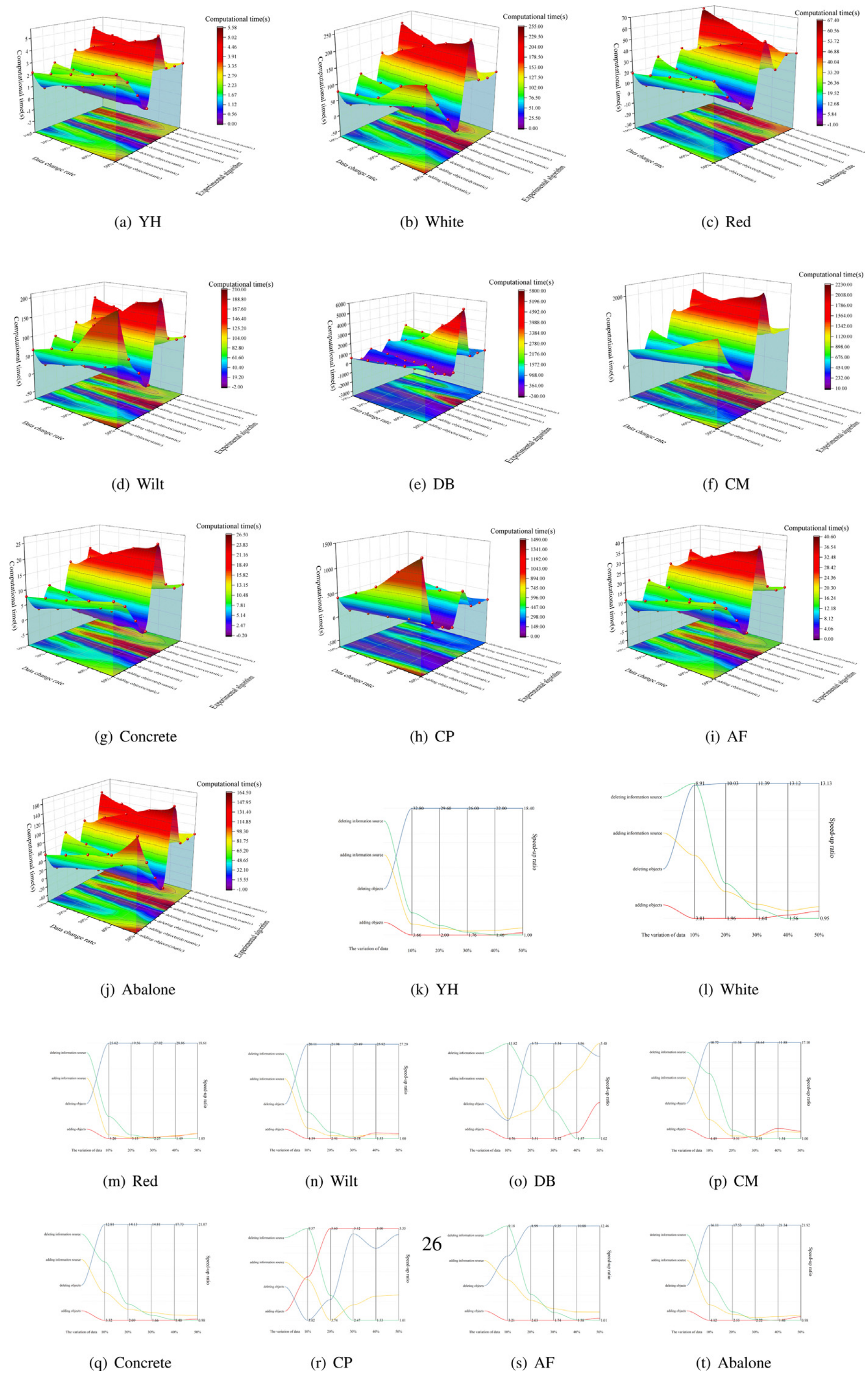

Fig. 11. Algorithm runtime and speedup for dataset.

(1) Comparison of static and dynamic algorithms when adding information sources. Figs. 11 $a-j$ and Table 9 present the results for the addition of information sources. As shown in Figs. 11 $a-j$ and Table 9, the runtime of the matrix-based dynamic updating approach and static algorithm increased as the number of information sources increased. The figure shows that the approach of updating the fixed aggregation operator with the dynamic matrix is substantially quicker than that of the static method. Figs. 11 $j-t$ demonstrate that after adding information sources, the speedup ratio between the dynamic and static approaches for each dataset is more than 1.

(2) Comparison of static and dynamic algorithms when deleting information sources. Fig. 11 and Table 10 present the results of deleting an information source. According to Fig. 11 $a-j$ and Table 10, with the reduction of information sources, the running time of the dynamic and static algorithms reduces to a certain extent. However, the running time of the dynamic algorithm remains shorter than that of the static algorithm. In addition, as shown in Figs. 11 $j-t$, the speedup ratio for most datasets is greater than 1. Although two datasets had a speedup ratio below 1 when 50% of the information sources were deleted, their speedup ratio remained near 1.

Table 9
Running time of each group of experiments when adding information source.

Datasets	10%		20%		30%		40%		50%	
	Static	Dynamic	Static	Dynamic	Static	Dynamic	Static	Dynamic	Static	Dynamic
Yach Hydrodynamics	3.28	0.53	3.86	1.09	4.34	1.70	4.86	2.17	5.55	2.80
Winequality-White	150.92	24.39	176.47	49.58	204.00	76.52	227.39	101.92	253.55	126.63
Winequality-Red	32.66	5.53	37.61	11.47	44.25	16.72	49.53	21.67	55.34	28.30
Wilt	110.20	17.94	131.70	37.52	151.42	56.44	190.75	73.48	208.16	103.33
Dry Bean	1282.17	204.84	1684.67	404.95	2312.66	603.86	3952.30	835.08	5822.11	1062.94
Crowdsourced Mapping	1352.73	236.45	1554.95	440.69	1789.88	655.38	1996.83	863.61	2223.03	1065.06
Concrete	15.80	2.61	18.53	5.30	21.30	7.83	23.92	10.47	26.31	12.95
Combined Cycle Power Plant	379.56	59.67	448.22	119.78	515.56	177.13	584.92	239.14	665.33	301.92
Airfoil Self Noise	24.31	4.22	28.23	8.05	32.20	12.02	36.06	16.20	40.27	20.00
Abalone	93.58	15.42	111.08	31.75	128.34	47.48	146.16	64.52	163.56	78.31

Table 10
Running time of each group of experiments when deleting information source.

Datasets	10%		20%		30%		40%		50%	
	Static	Dynamic	Static	Dynamic	Static	Dynamic	Static	Dynamic	Static	Dynamic
Yach Hydrodynamics	4.94	0.56	4.50	1.09	4.09	1.80	3.36	2.23	2.73	2.73
Winequality-White	229.06	25.70	200.69	49.45	172.73	75.06	151.39	96.83	119.09	125.25
Winequality-Red	68.78	7.27	60.73	16.20	53.28	23.47	45.31	30.48	39.20	38.06
Wilt	169.95	19.30	152.67	36.19	128.38	55.31	109.42	71.73	92.42	92.36
Dry Bean	2501.34	211.55	2098.78	419.92	1863.48	610.78	1293.45	823.50	1062.30	1043.86
Crowdsourced Mapping	1874.89	215.42	1663.08	413.44	1455.00	602.97	1240.16	804.53	1012.72	1011.03
Concrete	19.20	2.11	17.30	4.19	14.98	6.34	12.75	8.48	10.59	10.80
Combined Cycle Power Plant	576.92	60.28	502.28	118.30	434.95	176.28	360.05	235.02	299.39	297.84
Airfoil Self Noise	29.27	3.19	25.94	6.56	23.20	9.75	19.52	13.52	16.34	16.17
Abalone	144.70	16.17	126.63	32.58	112.73	47.48	96.69	66.08	77.58	79.47

Table 11
Speedup ratio for static and dynamic algorithms when information source change.

Variation ratio	Adding information source						Deleting information source					
	10%	20%	30%	40%	50%	Average	10%	20%	30%	40%	50%	Average
Yach Hydrodynamics	6.18	3.53	2.55	2.24	1.98	3.30	8.78	4.11	2.28	1.50	1.00	3.53
Winequality-White	6.19	3.56	2.67	2.23	2.00	3.33	8.91	4.06	2.30	1.56	0.95	3.56
Winequality-Red	5.90	3.28	2.65	2.29	1.96	3.21	9.47	3.75	2.27	1.49	1.03	3.60
Wilt	6.14	3.51	2.68	2.60	2.01	3.39	8.81	4.22	2.32	1.53	1.00	3.57
Dry Bean	6.26	4.16	3.83	4.73	5.48	4.89	11.82	5.00	3.05	1.57	1.02	4.49
Crowdsourced Mapping	5.72	3.53	2.73	2.31	2.09	3.28	8.70	4.02	2.41	1.54	1.00	3.54
Concrete	6.05	3.50	2.72	2.29	2.03	3.32	9.10	4.13	2.36	1.50	0.98	3.62
Combined Cycle Power Plant	6.36	3.74	2.91	2.45	2.20	3.53	9.57	4.25	2.47	1.53	1.01	3.76
Airfoil Self Noise	5.76	3.51	2.68	2.23	2.01	3.24	9.18	3.95	2.38	1.44	1.01	3.59
Abalone	6.07	3.50	2.70	2.27	2.09	3.32	8.95	3.89	2.37	1.46	0.98	3.53

Table 6 compares the speedup ratios of the static and dynamic methods at various ratios, illustrating the advantages of the dynamic algorithm when the information source changes. As shown in Table 11, the average speedups of the dynamic and static algorithms vary from 3.21 to 4.89 when adding information sources to the tests. Similarly, when information sources were deleted from the experiments, the average speedup varied from 3.53 to 4.49%. Figs. 11 $a-j$ and Table 11 imply that the dynamic approach is more computationally efficient than the static method. This finding is consistent with the theoretical analysis presented in Section 5.

6.3. Comparison with other relevant dynamic algorithms

Our study focused on the dynamic updating of multigranulation fusion operators in multisource information systems. However, existing dynamic and incremental algorithms have been primarily used to solve the dynamic update problem in single-source information systems. Hence, they cannot be directly applied to update the multigranulation fusion operator in a multi-source information system. Although no relevant dynamic algorithm has been developed for updating the multigranulation fusion operator values of objects or information sources that change over time, some dynamic algorithms can realize the dynamic update of single-source information systems. For a single information system, Zhang et al. proposed a matrix-based incremental algorithm to update the approximate value when adding or deleting objects (COV) [49]. For multisource information systems, Yang et al. proposed a method for directly obtaining multigranulation fusion operators from multisource information systems(CMGO) [48]. The combination of these two methods (COV + CMGO) can dynamically compute the multigranulation fusion operator when the objects of the multisource information system change over time. For information source changes, Zhang et al. investigated a dynamic algorithm that achieves the dynamic fusion of multi-source information systems (CIF) [35]. This method, combined with the computing multigranulation fusion operator approach, can update the multigranulation fusion operators under the change of information sources (CMGO+CIF). We compared these methods with the proposed algorithm on four cases, and Figs. 12 and 13 show the results. The sector size of $A(*)$ ($* =$ 10%, 20%, 30%, 40%, 50%) in Figs. 12 and 13 represents the running time of the proposed method when adding objects or information sources, and the sector size of $AC(*)$ represents the running time of the comparison algorithms. Sector size $D(*)$ represents the running time of the proposed method when deleting objects or information sources, and sector size $DC(*)$ represents the running time of the comparison algorithms. As shown in Fig. 12, when objects are added or deleted, the proposed dynamic

Fig. 12. Comparison of running time between the proposed dynamic algorithm and other relevant dynamic algorithms when objects are added or deleted.

algorithm has a faster calculation time than the COV + CMGO algorithm. As shown in Fig. 13, when an information source is added or deleted, the calculation times of the proposed dynamic algorithm and of the algorithm CMGO+CIF increase with the change ratio observed in Figs. 13 (a)C(h) and (j).

6.4. Statistical analysis

This study primarily focused on the matrix dynamic fusion mechanism of the multigranularity multisource information fusion approach when an object or information source changes. The speedup ratio is a useful indicator for comparing the execution times of dynamic and static algorithms. To verify the validity of the results, we conducted statistical tests and analyses on the speedup ratio of the dynamic and static algorithms. We conducted relevant statistical tests for adding or deleting objects and information sources. The speedup is the ratio between the running time of the dynamic and static algorithms. Therefore, we aimed to test whether the running time of the dynamic algorithm is better than that of the static algorithm, that is, to evaluate whether the speedup ratio is significantly larger than 1. Table 12

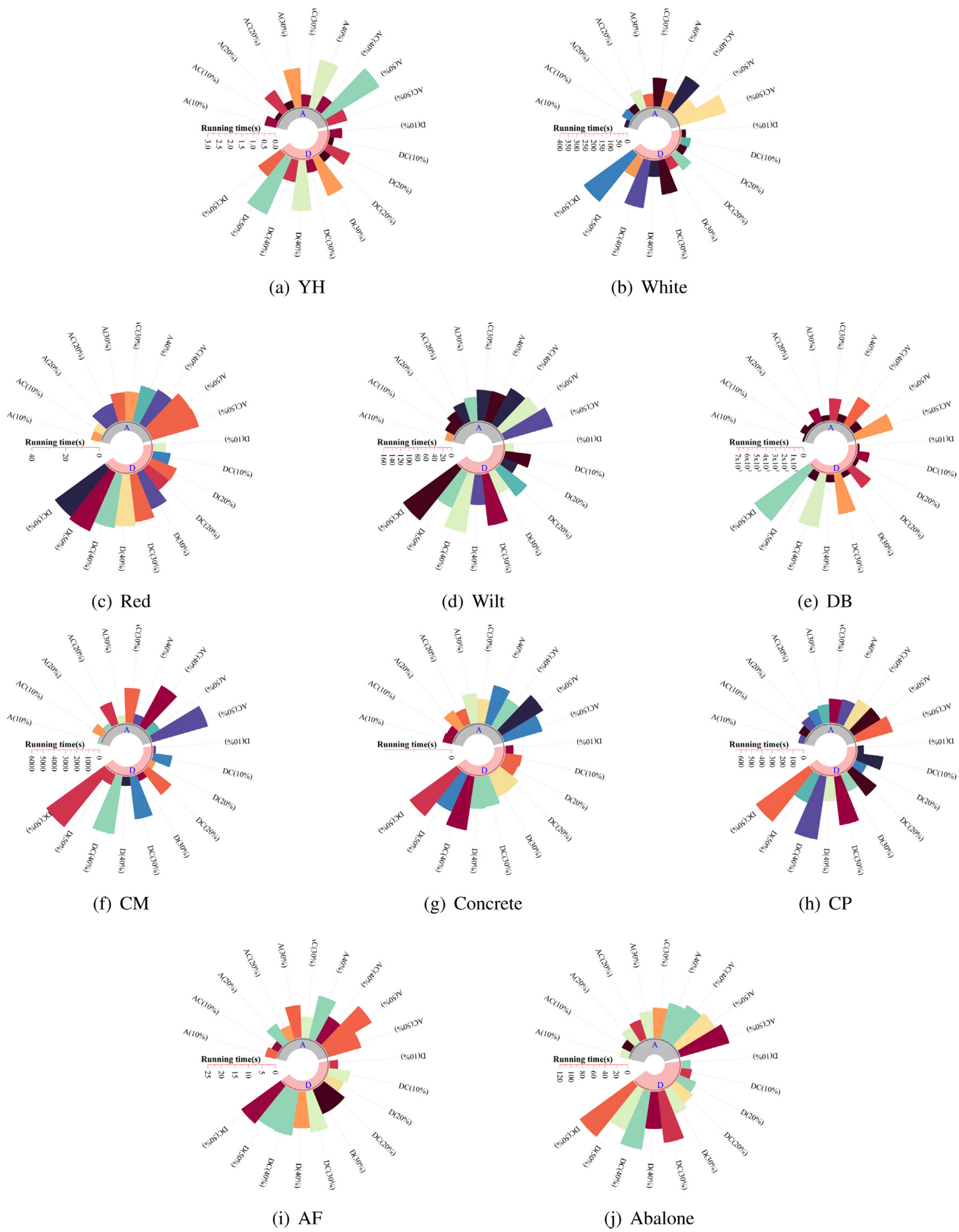

Fig. 13. Comparison of running time between the proposed dynamic algorithm and other relevant dynamic algorithms when information sources are added or deleted.

presents the statistical test results for the four cases. In addition, we provide boxplots of the speedup ratios for the four cases in Fig. 14.

The t-test, in addition to other statistical tests, was used to confirm that the average speedup was greater than 1. To more accurately demonstrate that the dynamic algorithm is superior to the static approach, we conducted a t-test to determine whether the speedup ratio was substantially greater than 2. As shown in Table 7, the *p*-values of the t-test were considerably less than 0.05 for all four cases, indicating that, with a 95% level of confidence, the dynamic algorithm has a much faster running time than the static algorithm. Concerning the boxplot, the horizontal axis indicates the proportion of multisource data changes in each experiment, and the vertical axis indicates the speedup ratio between the dynamic and static algorithms in each experiment. In addition, the four box plots represent four different experiments, that is, four dynamic changes in the multi-source data. The boxplot in Fig. 14 indicates that the general trend of the speedup

Table 12
Speedup ratio for static and dynamic algorithms when information source change.

	Adding objects	Deleting objects	Adding information source	Deleting information source
T-value	4.29	12.00	6.99	3.85
P-value	8.30×10^{-5}	3.37×10^{-16}	6.78×10^{-9}	3.37×10^{-4}

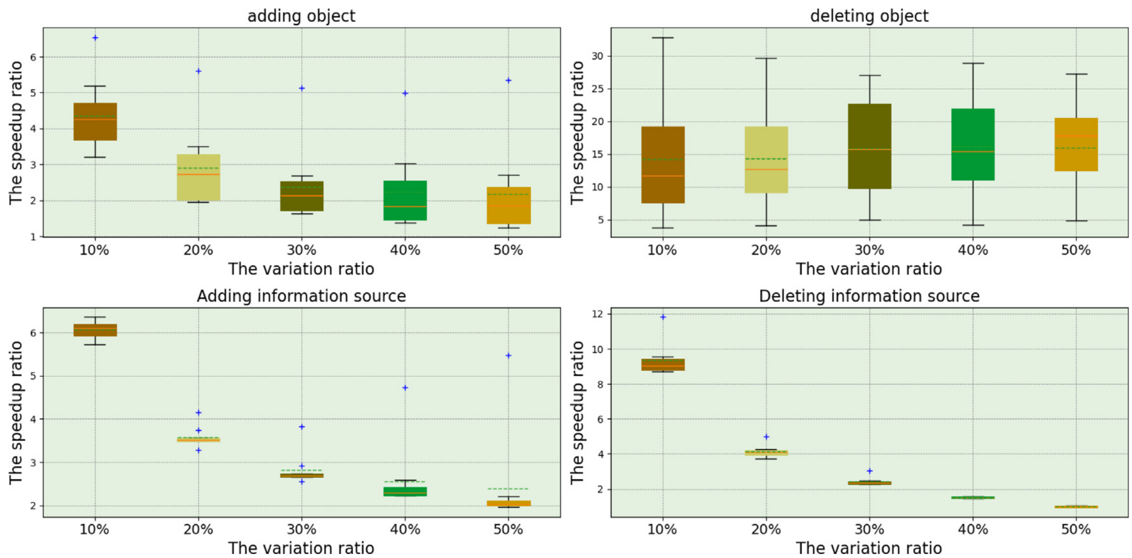

Fig. 14. Boxplots of speedup ratios for four cases.

ratio decreases as the change ratio increases, but remains greater than 1. Based on the statistical test results and boxplots, the dynamic method evidently outperforms the static approach in all four cases, which is consistent with the analyses presented in Sections 4 and 5.

7. Conclusion

This study investigated a strategy for dynamically updating multigranulation fusion operators. The primary contributions of this research are as follows. (1) We provide a matrix-based approach for calculating multigranulation information fusion operators. (2) For the four cases of adding objects, deleting objects, adding information sources, and deleting information sources, we constructed four dynamic algorithms to dynamically update fixed and possible aggregation operators based on a matrix. (3) Our evaluations show that the dynamic matrix-based algorithm for updating fixed and possible aggregation operators improves the computational efficiency compared with static matrix-based algorithms.

However, this study examined only dynamic changes in a single dimension of the multisource information system, thereby ignoring multidimensional changes. Moreover, an approach based on matrices has a relatively high spatial complexity. In the future, we plan to continue investigating the information fusion approach for multisource information systems with variations in attributes, objects, and information sources. Moreover, we can investigate dynamic algorithms with a lower space complexity.

CRediT authorship contribution statement

Xiaoyan Zhang: Conceptualization, Investigation, Methodology, Validation, Writing – review & editing. **Xudong Huang:** Data curation, Methodology, Software, Visualization, Writing – original draft, Writing – review & editing. **Weihua Xu:** Conceptualization, Funding acquisition, Investigation, Methodology, Project administration, Supervision, Validation.

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.

Data availability

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

Acknowledgments

The authors would like to thank the Associate Editor and the reviewers for their insightful comments and suggestions.

This paper is supported by the National Natural Science Foundation of China (No. 61976245). All authors approved the final version of the manuscript.

References

[1] Z. Pawlak, Rough sets, Int. J. Comput. Inform. Sci. 11 (5) (1982) 341–356.
[2] Z. Pawlak, Rough set theory and its applications to data analysis, Cybern. Syst. 29 (7) (1998) 661–688.
[3] Z. Pawlak, A. Skowron, Rough sets: Some extensions, Inform. Sci. 177 (1) (2007) 28–40.
[4] W. Ziarko, Variable precision rough set model, J. Comput. System Sci. 46 (1) (1993) 39–59.
[5] Y.Y. Yao, Probabilistic approaches to rough sets, Expert Syst. 20 (5) (2003) 287–297.
[6] Y.Y. Yao, Probabilistic rough set approximations, Internat. J. Approx. Reason. 49 (2) (2008) 255–271.
[7] R. Slowinski, D. Vanderpooten, A generalized definition of rough approximations based on similarity, IEEE Trans. Knowl. Data Eng. 12 (2) (2000) 331–336.
[8] Y.Y. Yao, Information granulation and rough set approximation, Int. J. Intell. Syst. 16 (1) (2001) 87–104.
[9] Y. Leung, M.M. Fischer, W.Z. Wu, et al., A rough set approach for the discovery of classification rules in interval-valued information systems, Internat. J. Approx. Reason. 47 (2) (2008) 233–246.

[10] W.S. Du, B.Q. Hu, Dominance-based rough set approach to incomplete ordered information systems, Inform. Sci. 346 (2016) 106–129.
[11] X.B. Yang, J.Y. Yang, Multigranulation rough sets in incomplete information system, in: Incomplete Information System and Rough Set Theory, Springer, 2012, pp. 195–222.
[12] J. Zhao, X.J. Xie, X. Xu, et al., Multi-view learning overview: Recent progress and new challenges, Inf. Fusion 38 (2017) 43–54.
[13] J.C. Zhang, D.Q. Zhang, A novel ensemble construction method for multi-view data using random cross-view correlation between within-class examples, Pattern Recognit. 44 (6) (2011) 1162–1171.
[14] Y.H. Qian, H. Zhang, Y.L. Sang, J.Y. Liang, Multigranulation decision-theoretic rough sets, Internat. J. Approx. Reason. 55 (1) (2014) 225–237.
[15] W.H. Xu, W.T. Li, Granular computing approach to two-way learning based on formal concept analysis in fuzzy datasets, IEEE Trans. Cybern. 46 (2) (2014) 366–379.
[16] P.F. Zhang, T.R. Li, C. Luo, G.Q. Wang, AMG-DTRS: Adaptive multi-granulation decision-theoretic rough sets, Internat. J. Approx. Reason. 140 (2022) 7–30.
[17] G.P. Lin, J.Y. Liang, Y.H. Qian, An information fusion approach by combining multigranulation rough sets and evidence theory, Inform. Sci. 314 (2015) 184–199.
[18] G.P. Lin, J.Y. Liang, Y.H. Qian, et al., A fuzzy multigranulation decision-theoretic approach to multi-source fuzzy information systems, Knowl.-Based Syst. 91 (2016) 102–113.
[19] W.H. Xu, J.H. Yu, A novel approach to information fusion in multi-source datasets: A granular computing viewpoint, Inform. Sci. 378 (2017) 410–423.
[20] W.H. Xu, M.M. Li, X.Z. Wang, Information fusion based on information entropy in fuzzy multi-source incomplete information system, Int. J. Fuzzy Syst. 19 (4) (2017) 1200–1216.
[21] C.X. Hu, S.X. Liu, G.X. Liu, Matrix-based approaches for dynamic updating approximations in multigranulation rough sets, Knowl.-Based Syst. 122 (2017) 51–63.
[22] T.R. Li, C. Luo, H.M. Chen, J.B. Zhang, The principles and methodologies of big data miningł-From the perspectives of granular computing and rough sets, 2016.
[23] T.R. Li, C. Luo, H.M. Chen, J.B. Zhang, PICKT: A solution for big data analysis, in: International Conference on Rough Sets and Knowledge Technology, Springer, 2015, pp. 15–25.
[24] P.F. Zhang, T.R. Li, G.Q. Wang, C. Luo, H.M. Chen, J.B. Zhang, D.X. Wang, Z. Yu, Multi-source information fusion based on rough set theory: A review, Inform. Fusion 68 (2021) 85–117.
[25] C.X. Hu, L. Zhang, B.J. Wang, Z. Zhang, F.Z. Li, Incremental updating knowledge in neighborhood multigranulation rough sets under dynamic granular structures, Knowl.-Based Syst. 163 (2019) 811–829.
[26] J.B. Laszczyski, R.S. Lowiski, Incremental induction of decision rules from dominance-based rough approximations, Electron. Notes Theor. Comput. Sci. 82 (4) (2003) 40–51.
[27] C. Luo, T.R. Li, H.M. Chen, D. Liu, Incremental approaches for updating approximations in set-valued ordered information systems, Knowl.-Based Syst. 50 (2013) 218–233.
[28] W.T. Li, W.H. Xu, X.Y. Zhang, J. Zhang, Updating approximations with dynamic objects based on local multigranulation rough sets in ordered information systems, Artif. Intell. Rev. 55 (3) (2022) 1821–1855.
[29] S.Y. Li, T.R. Li, D. Liu, Dynamic maintenance of approximations in dominance-based rough set approach under the variation of the object set, Int. J. Intell. Syst. 28 (8) (2013) 729–751.
[30] W.H. Shu, W.B. Qian, Y.H. Xie, Incremental approaches for feature selection from dynamic data with the variation of multiple objects, Knowl.-Based Syst. 163 (2019) 320–331.
[31] Y.Y. Yang, D.G. Chen, H. Wang, Active sample selection based incremental algorithm for attribute reduction with rough sets, IEEE Trans. Fuzzy Syst. 25 (4) (2016) 825–838.
[32] C.X. Hu, L. Zhang, A dynamic framework for updating neighborhood multigranulation approximations with the variation of objects, Inform. Sci. 519 (2020) 382–406.
[33] H.M. Chen, T.R. Li, H.M. Tian, Approaches for updating approximations in set-valued information systems while objects and attributes vary with time, in: Rough Sets and Intelligent Systems-Professor Zdzislaw Pawlak in Memoriam, Springer, 2013, pp. 229–248.
[34] F. Wang, J.Y. Liang, Y.H. Qian, Attribute reduction: A dimension incremental strategy, Knowl.-Based Syst. 39 (2013) 95–108.
[35] X.Y. Zhang, J.R. Li, J.S. Mi, Dynamic updating approximations approach to multi-granulation interval-valued hesitant fuzzy information systems with time-evolving attributes, Knowl.-Based Syst. 238 (2022) 107809.
[36] A.P. Zeng, T.R.Li, D. Liu, J.B. Zhang, H.M. Chen, A fuzzy rough set approach for incremental feature selection on hybrid information systems, Fuzzy Sets and Systems 258 (2015) 39–60.
[37] C.X. Hu, L. Zhang, Incremental updating probabilistic neighborhood three-way regions with time-evolving attributes, Internat. J. Approx. Reason. 120 (2020) 1–23.
[38] Q.Q. Huang, T.R. Li, Y.Y. Huang, X. Yang, H. Fujita, Dynamic dominance rough set approach for processing composite ordered data, Knowl.-Based Syst. 187 (2020) 104829.
[39] C. Luo, T.R. Li, H.M. Chen, et al., Fast algorithms for computing rough approximations in set-valued decision systems while updating criteria values, Inform. Sci. 299 (2015) 221–242.
[40] Y.Y. Huang, T.R. Li, C. Luo, H. Fujita, S.J. Horng, B. Wang, Dynamic maintenance of rough approximations in multi-source hybrid information systems, Inform. Sci. 530 (2020) 108–127.
[41] Y.G. Jing, T.R. Li, J.F. Huang, H.M. Chen, S.J. Horng, A group incremental reduction algorithm with varying data values, Int. J. Intell. Syst. 32 (9) (2017) 900–925.
[42] W. Wei, X.Y. Wu, J.Y. Liang, J.B. Cui, Y.J. Sun, Discernibility matrix based incremental attribute reduction for dynamic data, Knowl.-Based Syst. 140 (2018) 142–157.
[43] R.K. Dubey, S.S. Sohn, C. Hoelscher, M. Kapadia, Fusion-based wayfinding prediction model for multiple information sources, in: In 2019 22th IEEE International Conference on Information Fusion, 2019, pp. 1–8.
[44] Y.Y. Huang, K.J. Guo, X.W. Yi, Z. Li, T.R. Li, Matrix representation of the conditional entropy for incremental feature selection on multi-source data, Inform. Sci. 591 (2022) 263–286.
[45] W.H. Xu, Y.Z. Pan, X.W. Chen, Y.H. Qian, W.P. Ding, A novel dynamic fusion approach using information entropy for interval-valued ordered datasets, IEEE Trans. Big Data (2022) 321549, http://dx.doi.org/10.1109/TBDATA.2022.321549.
[46] X.Y. Zhang, X.W. Chen, W.H. Xu, et al., Dynamic information fusion in multi-source incomplete interval-valued information system with variation of information sources and attributes, Inform. Sci. 608 (2022) 1–27.
[47] W.H. Xu, Order Information System and Rough Set, Science Press, Beijing, 2013.
[48] L. Yang, W.H. Xu, X.Y. Zhang, B.B. Sang, Multi-granulation method for information fusion in multi-source decision information system, Internat. J. Approx. Reason. 122 (2020) 47–65.
[49] J.B. Zhang, T.R. Li, H.M. Chen, Composite rough sets for dynamic data mining, Inform. Sci. 257 (2014) 81–100.