

# Decision Rule Aggregation Approach to Support Group Decision Making<sup>\*</sup>

Inès Saad<sup>1,2</sup> and Salem Chakhar<sup>3</sup>

<sup>1</sup> MIS, University of Picardie Jules Verne, Amiens, France [ines.saad@u-picardie.fr](mailto:ines.saad@u-picardie.fr)

<sup>2</sup> Amiens Business School, Amiens, France

<sup>3</sup> Portsmouth Business School & CORL, University of Portsmouth, Portsmouth, UK  
[salem.chakhar@port.ac.uk](mailto:salem.chakhar@port.ac.uk)

**Abstract.** The Dominance-based Rough Set Approach (DRSA) is an innovative preference learning approach. It takes as input a set of objects (learning set) described with respect to a collection of condition and decision attributes. It generates a set of if-then decision rules. Initial versions of dominance based rough set approximation methods assume a single decision maker. Furthermore, the proposed extensions to group decision making mainly use an input oriented aggregation strategy, which requires a high level of agreement between the decision makers. In this paper, we propose an output oriented aggregation strategy to coherently combine different sets of decision rules obtained from different decision makers. The proposed aggregation algorithm is illustrated by using real-world data relative to a business school admission where two decision makers are involved. Results show that aggregation algorithm is able to reproduce the individual assignments of students with a very limited preferential information loss.

**Keywords:** Rough set approximation · Decision rule · Group decision making · Rules aggregation · Output aggregation strategy

## 1 Introduction

The Dominance-based Rough Set Approach (DRSA) [7] is an extension of the Rough Sets Theory [10] intended to deal with multicriteria sorting problems. The DRSA takes a set of assignment examples (learning set) and generates a collection of if-then decision rules as output. The conventional DRSA assumes a single decision maker while several real-world decision problems need to take into account the presence of multiple decision makers. Different group decision making extensions to DRSA have been proposed in the literature, including [1][2][3][4][5][6][8][11][12].

For instance, the authors in [1] and [8] extend the concepts of the DRSA to deal with decision tables having multiple decision attributes, thus allowing comprehensive collective decision rules to be generated. In [4] we introduced an

---

<sup>\*</sup> [Post print version, please cite as] Inès Saad, Salem Chakhar (2020). Decision Rule Aggregation Approach to Support Group Decision Making. In: Proceedings of the 20th International Conference on Group Decision and Negotiation (GDN 2020), Lecture Notes in Business Information Processing.

aggregation algorithm, based on the majority principle and supporting the veto effect, allowing consensual decision rules to be inferred. A more advanced version of the aggregation algorithm of [4] is proposed in [3]. In [5][6], the authors use the Dempster-Shafer theory of evidence to combine individual rules provided by the DRSA.

However, all these approaches rely on an input oriented aggregation strategy, which requires a high level of agreement between the decision makers. In this paper, we propose an output oriented aggregation strategy to coherently combine different sets of decision rules obtained from different decision makers. The proposed aggregation algorithm is illustrated by using real-world data relative to business school admission where two decision makers are involved. Results show that aggregation algorithm is able to reproduce the individual assignments of students with a very low preferential information loss.

The rest of the paper is structured as follows. Section 2 sets the background. Section 3 deals with rules matching and overleaping. Section 4 details the aggregation algorithm. Section 5 provides an illustrative application. Section 6 concludes the paper.

## 2 Background

### 2.1 Notations and Basic Assumptions

Information about decision objects is often represented in terms of an *information table* where rows correspond to *objects* and columns to *attributes*. The information table  $S$  is a 4-tuple  $\langle U, Q, V, f \rangle$  where:  $U$  is a finite set of objects,  $Q$  is a finite set of attributes,  $V = \bigcup_{q \in Q} V_q$ , where  $V_q$  is a domain of the attribute  $q$ , and  $f : U \times Q \rightarrow V$  an *information function* defined such that  $f(x, q) \in V_q, \forall q \in Q, \forall x \in U$ . The set of attributes  $Q$  is often divided into a sub-set  $C$  of *condition attributes* and a sub-set  $D$  of *decision attributes*. In this case,  $S$  is called *decision table*.

The domain of condition attributes is supposed to be ordered in decreasing or increasing preference. Such attributes are called *criteria*. We assume that the preference is increasing with value of  $f(\cdot, q)$  for every  $q \in C$ . We also assume that the set of decision attributes  $D$  is a singleton  $\{d\}$ . Decision attribute  $d$  makes a partition of  $U$  into a finite number of decision classes  $\mathbf{CI} = \{Cl_t, t \in T\}$ ,  $T = \{0, \dots, n\}$ , such that each  $x \in U$  belongs to one and only one class in  $\mathbf{CI}$ . Further, we assume that the classes are preference-ordered, i.e. for all  $r, s \in T$ , such that  $r > s$ , the objects from  $Cl_r$  are preferred to the objects from  $Cl_s$ .

### 2.2 Rough Approximation

In DRSA the represented knowledge is a collection of *upward unions*  $Cl_t^{\geq}$  and *downward unions*  $Cl_t^{\leq}$  of classes defined as follows:  $Cl_t^{\geq} = \cup_{s \geq t} Cl_s$  and  $Cl_t^{\leq} = \cup_{s \leq t} Cl_s$ . The assertion “ $x \in Cl_t^{\geq}$ ” means that “ $x$  belongs to at least class  $Cl_t$ ” while assertion “ $x \in Cl_t^{\leq}$ ” means that “ $x$  belongs to at most class  $Cl_t$ ”. The basic

idea of DRSA is to replace the indiscernibility relation used in the conventional Rough Set Theory with a dominance relation. Let  $P \subseteq C$  be a subset of condition attributes. The *dominance relation*  $\Delta_P$  associated with  $P$  is defined for each pair of objects  $x$  and  $y$  as follows:

$$x\Delta_P y \Leftrightarrow f(x, q) \succeq f(y, q), \forall q \in P. \quad (1)$$

In the definition above, the symbol “ $\succeq$ ” should be replaced with “ $\preceq$ ” for condition attributes which are ordered according to decreasing preferences. To each object  $x \in U$ , we associate two sets: (i) the *P-dominating set*  $\Delta_P^+(x) = \{y \in U : y\Delta_P x\}$  containing the objects that dominate  $x$ , and (ii) the *P-dominated set*  $\Delta_P^-(x) = \{y \in U : x\Delta_P y\}$  containing the objects dominated by  $x$ .

Then, the *P-lower* and *P-upper* approximations of  $Cl_t^{\succeq}$  with respect to  $P$  are defined as follows:

- $\underline{P}(Cl_t^{\succeq}) = \{x \in U : \Delta_P^+(x) \subseteq Cl_t^{\succeq}\},$
- $\bar{P}(Cl_t^{\succeq}) = \{x \in U : \Delta_P^-(x) \cap Cl_t^{\succeq} \neq \emptyset\}.$

Analogously, the *P-lower* and *P-upper* approximations of  $Cl_t^{\preceq}$  with respect to  $P$  are defined as follows:

- $\underline{P}(Cl_t^{\preceq}) = \{x \in U : \Delta_P^-(x) \subseteq Cl_t^{\preceq}\},$
- $\bar{P}(Cl_t^{\preceq}) = \{x \in U : \Delta_P^+(x) \cap Cl_t^{\preceq} \neq \emptyset\}.$

The lower approximations group the objects which certainly belong to class unions  $Cl_t^{\succeq}$  (resp.  $Cl_t^{\preceq}$ ). The upper approximations group the objects which could belong to  $Cl_t^{\succeq}$  (resp.  $Cl_t^{\preceq}$ ).

The *P-boundaries* of  $Cl_t^{\succeq}$  and  $Cl_t^{\preceq}$  are defined as follows:

- $Bn_P(Cl_t^{\succeq}) = \bar{P}(Cl_t^{\succeq}) - \underline{P}(Cl_t^{\succeq}),$
- $Bn_P(Cl_t^{\preceq}) = \bar{P}(Cl_t^{\preceq}) - \underline{P}(Cl_t^{\preceq}).$

The boundaries group objects that can neither be ruled in nor out as members of class  $Cl_t$ .

### 2.3 Decision Rules

The approximations of upward and downward unions of classes can serve to induce a set of if-then decision rules relating condition and decision attributes. There are five basic types of decision rules:

- *Certain decision rules.* These rules are generated from the lower approximation of the union of classes  $Cl_t^{\preceq}$  or  $Cl_t^{\succeq}$ . A decision rule from this type has one of the following structures:
  - **Type 1:** if  $f(x, q_1) \leq r_1 \wedge \dots \wedge f(x, q_m) \leq r_m$  then  $x \in Cl_t^{\preceq}$
  - **Type 2:** if  $f(x, q_1) \geq r_1 \wedge \dots \wedge f(x, q_m) \geq r_m$  then  $x \in Cl_t^{\succeq}$
 where  $(r_1, \dots, r_m) \in (V_{q_1} \times \dots \times V_{q_m})$ .

- *Possible decision rules.* These rules are generated from the upper approximation of the union of classes  $Cl_t^{\leq}$  or  $Cl_t^{\geq}$ . A decision rule from this type has one of the following structures:
  - **Type 3:** if  $f(x, q_1) \leq r_1 \wedge \dots \wedge f(x, q_m) \leq r_m$  **then**  $x$  could belong to  $Cl_t^{\leq}$
  - **Type 4:** if  $f(x, q_1) \geq r_1 \wedge \dots \wedge f(x, q_m) \geq r_m$  **then**  $x$  could belong to  $Cl_t^{\geq}$
 where  $(r_1, \dots, r_m) \in (V_{q_1} \times \dots \times V_{q_m})$ .
- *Approximate rules.* These rules are generated from the boundaries. A decision rule from this type has the following structure:
  - **Type 5:** if  $f(x, q_1) \leq r_1 \wedge \dots \wedge f(x, q_m) \leq r_m \wedge f(x, q_{m+1}) \leq r_{m+1} \wedge \dots \wedge f(x, r_p) \leq r_p$  **then**  $x \in Cl_s \cup Cl_{s+1} \cup \dots \cup Cl_t$
 where  $(r_1, \dots, r_p) \in (V_{q_1} \times \dots \times V_{q_p})$ .

Only the two first types are considered in the rest of this paper.

The most popular rule induction algorithm for DRSA is DOMLEM [9], which generates a minimal set of rules.

### 3 Decision Rules Matching and Overlapping

#### 3.1 Basic Definitions

A decision rule  $R$  is defined as a collection of elementary conditions and a conclusion. Let  $R.C$  denotes the set of conditions of rule  $R$  and  $R.C_i$  denote each member of this set. Let  $R.N$  denote the cardinality of this set. Let  $R.D$  denotes the conclusion associated with rule  $R$ . Each decision rule  $R$  is characterized by its type  $R.T$ .

An elementary condition  $C_i$  is defined by an attribute  $Q$ , an operator  $O$  and a right-hand member  $H$ . Let  $C_i.Q$ ,  $C_i.O$  and  $C_i.H$  denote these three elements.

#### 3.2 Conditions Matching

**Definition 1 (Conditions equality).** Let  $C_k$  and  $C_l$  be two conditions of the same type. Then,  $C_k$  is equal to  $C_l$  (denoted  $C_k = C_l$ ) iff:

$$\left\{ \begin{array}{l} C_k.Q = C_l.Q \\ C_k.O = C_l.O \\ C_k.H = C_l.H \end{array} \right\} \Rightarrow (C_k = C_l)$$

**Definition 2 (Type 1 conditions inclusion).** Let  $C_k$  and  $C_l$  be two conditions of Type 1 decision rules. Then,  $C_k$  is included in  $C_l$  (denoted  $C_k \subseteq C_l$ ) iff:

$$\left\{ \begin{array}{l} C_k.Q = C_l.Q \\ C_k.O = C_l.O \\ C_k.H \preceq C_l.H \end{array} \right\} \Rightarrow (C_k \subseteq C_l)$$

**Definition 3 (Type 2 conditions inclusion).** Let  $C_k$  and  $C_l$  be two conditions of Type 2 decision rules. Then,  $C_k$  includes  $C_l$  (denoted  $C_k \supseteq C_l$ ) iff:

$$\left\{ \begin{array}{l} C_k.Q = C_l.Q \\ C_k.O = C_l.O \\ C_k.H \succeq C_l.H \end{array} \right\} \Rightarrow (C_k \supseteq C_l)$$

### 3.3 Decision Rules Matching

The equality between two decision rules is defined as follows.

**Definition 4 (Decision rules equality).** Let  $R_i$  and  $R_j$  be two decision rules of the same type. Then,  $R_i$  is equal to  $R_j$  (denoted  $R_i = R_j$ ) iff:

$$\left\{ \begin{array}{l} R_i.T = R_j.T \\ \forall k \exists l (R_i.C_k = R_j.C_l) \quad 1 \leq k \leq R_i.N \\ \forall m \exists n (R_j.C_m = R_i.C_n) \quad 1 \leq m \leq R_j.N \\ R_i.D = R_j.D \end{array} \right\} \Rightarrow (R_i = R_j)$$

**Definition 5 (Decision rules Type 1 full inclusion).** Let  $R_i$  and  $R_j$  be two decision rules of Type 1. Then,  $R_i$  is fully included in  $R_j$  (denoted  $R_i \subseteq R_j$ ) iff:

$$\left\{ \begin{array}{l} R_i.T = R_j.T = \text{Type 1} \\ \forall k \exists l (R_i.C_k \subseteq R_j.C_l) \quad 1 \leq k \leq R_i.N \\ \forall m \exists n (R_i.C_n \subseteq R_j.C_m) \quad 1 \leq m \leq R_i.N \\ R_i.D \preceq R_j.D \end{array} \right\} \Rightarrow (R_i \subseteq R_j)$$

This definition implicitly ensures that rules  $R_i$  and  $R_j$  must have the same cardinality, i.e.,  $R_i.N = R_j.N$ .

**Definition 6 (Decision rules Type 1 Partial inclusion).** Let  $R_i$  and  $R_j$  be two decision rules of Type 1. Then,  $R_i$  is partially included in  $R_j$  (denoted  $R_i \subset R_j$ ) iff:

$$\left\{ \begin{array}{l} R_i.T = R_j.T = \text{Type 1} \\ \forall k \exists l (R_i.C_k \subseteq R_j.C_l) \quad 1 \leq k \leq R_i.N \\ R_i.D \preceq R_j.D \\ R_i.N < R_j.N \end{array} \right\} \Rightarrow (R_i \subset R_j)$$

The last condition (i.e.,  $R_i.N < R_j.N$ ) in this definition ensures that the cardinality of rule  $R_i$  must be strictly less than the one of  $R_j$ .

**Definition 7 (Decision rules Type 2 full inclusion).** Let  $R_i$  and  $R_j$  be two decision rules of Type 2. Then,  $R_i$  is fully included in  $R_j$  (denoted  $R_i \supseteq R_j$ ) iff:

$$\left\{ \begin{array}{l} R_i.T = R_j.T = \text{Type 2} \\ \forall k \exists l (R_i.C_k \supseteq R_j.C_l) \quad 1 \leq k \leq R_i.N \\ \forall m \exists n (R_i.C_n \subseteq R_j.C_m) \quad 1 \leq m \leq R_i.N \\ R_i.D \succeq R_j.D \end{array} \right\} \Rightarrow (R_i \supseteq R_j)$$

This definition implicitly ensures that rules  $R_i$  and  $R_j$  must have the same cardinality, i.e.,  $R_i.N = R_j.N$ .

**Definition 8 (Decision rules Type 2 partial inclusion).** *Let  $R_i$  and  $R_j$  be two decision rules of Type 2. Then,  $R_i$  is partially included in  $R_j$  (denoted  $R_i \supset R_j$ ) iff:*

$$\left\{ \begin{array}{l} R_i.T = R_j.T = \text{Type 2} \\ \forall k \exists l (R_i.C_k \supseteq R_j.C_l) \quad 1 \leq k \leq R_i.N \\ R_i.D \succeq R_j.D \\ R_i.N < R_j.N \end{array} \right\} \Rightarrow (R_i \supset R_j)$$

The last condition (i.e.,  $R_i.N < R_j.N$ ) in this definition ensures that the cardinality of rule  $R_i$  must be strictly less than the one of  $R_j$ .

### 3.4 Overlapping Decision Rules

Let  $R_i$  be a Type 1 decision rule and  $R_j$  be a Type 2 decision rule. Although these rules are of different types, they may or not share some parts of their conditions and/or decisions. Four basic cases can be distinguished: (i)  $R_i$  and  $R_j$  are fully disjoint; (ii)  $R_i$  and  $R_j$  have overlapped conditions but their decisions are disjoint; (iii)  $R_i$  and  $R_j$  have overlapped decisions but their conditions are disjoint; and (iv)  $R_i$  and  $R_j$  have overlapped conditions and overlapped decisions.

**Definition 9 (Decision rules with overlapped conditions).** *Let  $R_i$  be a Type 1 decision rule and  $R_j$  be a Type 2 decision rule. Decision rules  $R_i$  and  $R_j$  have overlapped conditions, denoted  $(R_i.C \cap R_j.C) \neq \emptyset$ , iff:*

$$\left\{ \begin{array}{l} \forall C_k \in R_i.C, \exists C_l \in R_j.C (C_k.Q = C_l.Q) \wedge \\ (C_k.O \neq C_l.O) \wedge (C_k.H \geq C_l.H) \quad 1 \leq k \leq R_i.N \\ \forall C_m \in R_j.C, \exists C_n \in R_i.C (C_n.Q = C_m.Q) \wedge \\ (C_n.O \neq C_m.O) \wedge (C_m.H \leq C_n.H) \quad 1 \leq m \leq R_j.N \end{array} \right\} \Rightarrow (R_i.C \cap R_j.C) \neq \emptyset$$

**Definition 10 (Decision rules with overlapped decisions).** *Let  $R_i$  be a Type 1 decision rule and  $R_j$  be a Type 2 decision rule. Decision rules  $R_i$  and  $R_j$  have overlapped decisions, denoted  $(R_i.D \cap R_j.D) \neq \emptyset$ , iff only if:  $R_i.D \preceq R_j.D$ .*

**Definition 11 (Fully overlapped decision rules).** *Let  $R_i$  be a Type 1 decision rule and  $R_j$  be a Type 2 decision rule. Decision rules  $R_i$  and  $R_j$  are fully overlapped, denoted  $(R_i.C \cap R_j.C) \neq \emptyset$ , iff:*

$$\left\{ \begin{array}{l} \forall C_k \in R_i.C, \exists C_l \in R_j.C (C_k.Q = C_l.Q) \wedge \\ (C_k.O \neq C_l.O) \wedge (C_k.H \geq C_l.H) \\ \forall C_m \in R_j.C, \exists C_n \in R_i.C (C_n.Q = C_m.Q) \wedge \\ (C_n.O \neq C_m.O) \wedge (C_m.H \leq C_n.H) \\ R_i.D \succeq R_j.D \end{array} \right\} \Rightarrow (R_i \cap R_j) \neq \emptyset$$

**Definition 12 (Disjoint decision rules).** Let  $R_i$  be a Type 1 decision rule and  $R_j$  be a Type 2 decision rule. Decision rules  $R_i$  and  $R_j$  are disjoint, denoted  $(R_i \cap R_j) = \emptyset$ , iff:

$$\left\{ \begin{array}{l} \forall C_k \in R_i.C, \exists C_l \in R_j (C_k.Q = C_l.Q) \wedge \\ \quad (C_k.O \neq C_l.O) \wedge (C_k.H < C_l.H) \\ \forall C_l \in R_j.C, \exists C_k \in R_i (C_l.Q = C_k.Q) \wedge \\ \quad (C_l.O \neq C_k.O) \wedge (C_l.H > C_k.H) \\ R_i.D \prec R_j.R \end{array} \right\} \Rightarrow (R_i \cap R_j) = \emptyset$$

## 4 Decision Rules Aggregation

Let  $H = \{1, \dots, i, \dots, h\}$  be a set of decision makers and  $\Pi_i$  the set decision rules obtained by decision maker  $i \in H$ . Let  $\Pi$  be the union of all decision rules of the  $h$  decision makers:  $\Pi = \cup_{i=1}^h \Pi_i$ .

The aggregation algorithm contains two steps: (i) transformation of overlapping rules, and (ii) elimination of redundant decision rules. We should mention that steps (i) and (ii) may be inverted without affecting the final result. However, in this paper, we maintain the order given above for several reasons. First, the other solution (i.e. proceeding by computing the minimal cover and then transformation of overlapping rules) requires an additional step to compute the minimal cover after the transformation operation. Indeed, the latter may lead to new redundant rules. Second, as a consequence of the first point, the computing time will automatically increase. The only shortcoming of the solution adopted in this paper is that in step (i) both redundant and non-redundant rules are considered. This may have minor effects on the overall computing time.

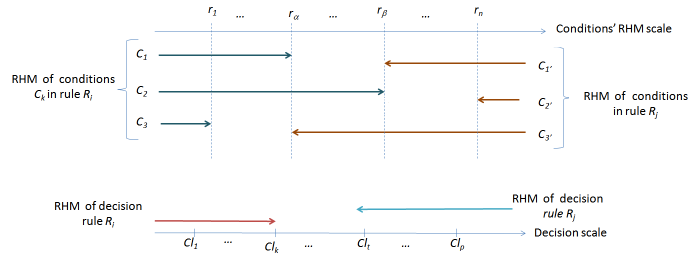
The aggregation algorithm takes the set  $\Pi$  of all decision rules and generates a minimal set of non-redundant decision rules.

### 4.1 Step 1: Transformation of Overlapping Decision Rules

**Case 1.  $R_i$  and  $R_j$  are disjoint decision rules** This situation is graphically illustrated by Figure 1. In this figure, we assumed that all conditions attributes have the same scale and that  $1 < \alpha < \beta < n$ . As it is shown in Figure 1, the constraints defined by the conditions of rules  $R_i$  and  $R_j$  are totally disjoint. For instance, condition  $C_2$  says that  $f(x, A_s) \leq r_\beta$  and condition  $C_{2'}$  says that  $f(x, A_s) \geq r_n$ . It is easy to see that there is no any intersection between the two constraints as defined by  $C_2$  and  $C_{2'}$ . The same remark holds for the other conditions and for the decision of  $R_i$  and  $R_j$ .

In this situation, there is no overlap between decision rules  $R_i$  and  $R_j$  and it is reasonable to maintain both of them (if they are not overlapped by other rules).

**Case 2.  $R_i$  and  $R_j$  have overlapped conditions and decisions** This situation is graphically illustrated by Figure 2. In this figure, we assumed that all



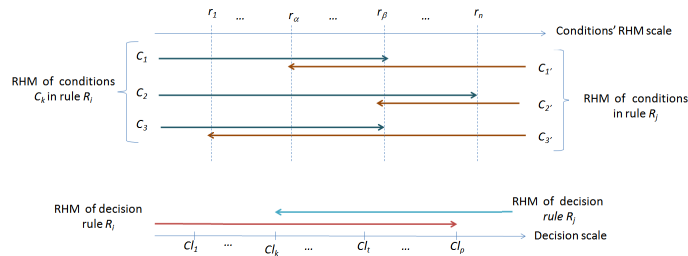
**Fig. 1.** Schematic representation of disjoint decision rules.

conditions attributes have the same scale and that  $1 < \alpha < \beta < n$ . As it is shown in Figure 2, the constraints defined by the conditions of rules  $R_i$  and  $R_j$  overlap. For instance, condition  $C_2$  says that  $f(x, A_s) \leq r_n$  and condition  $C_{2'}$  says that  $f(x, A_s) \geq r_\beta$ . It is easy to see that there is an intersection between the two constraints defined by  $C_2$  and  $C_{2'}$  (since  $\beta < n$ ). The same remark holds for the other conditions. The same remark holds for decisions of  $R_i$  and  $R_j$ .

To reduce the interval-based assignments of decision objects, we propose to replace rules  $R_i$  and  $R_j$  by three fully disjoint decision rules as follows:

- $R_a$ : with the same structure and type as rule  $R_i$  but the Right Hand Side (RHS) of conditions and the decision are those of rule  $R_j$ ;
- $R_b$ : with the same structure and type as rule  $R_j$  but the RHS of conditions and the decision are those of rule  $R_i$ ;
- $R_c$ : the RHS of conditions are of the form  $[R_j.C_k.H, R_i.C_k.H]$  and the decision is of the form  $[R_j.D, R_i.D]$ .

The last decision rule is of composite type since RHS of the conditions and the decision are interval-based.



**Fig. 2.** Schematic representation of full overlap of decision rules

**Case 3.  $R_i$  and  $R_j$  have overlapped conditions** This situation is graphically illustrated by Figure 3. In this figure, we assume that all conditions attributes



have the same scale and that  $1 < \alpha < \beta < n$ . As it is shown in Figure 3, the constraints defined by the conditions of rules  $R_i$  and  $R_j$  overlap but the decisions do not. For instance, condition  $C_2$  says that  $f(x, A_s) \leq r_n$  are condition  $C_{2'}$  says that  $f(x, A_s) \geq r_\beta$ . It is easy to see that there is any intersection between the two constraints defined by  $C_2$  and  $C_{2'}$  (since  $\beta < n$ ). The same remark holds for the other conditions. On the contrary, the decision parts of rules  $R_i$  and  $R_j$  are totally disjoint.

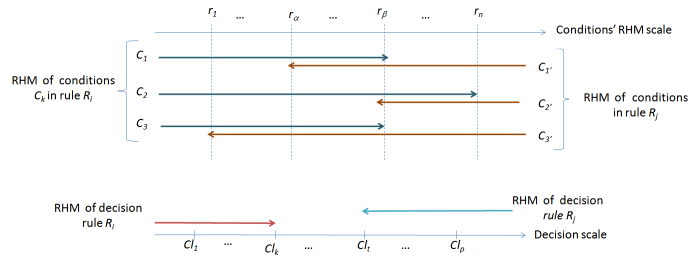
To reduce the interval-based assignments of decision objects, we propose to replace rules  $R_i$  and  $R_j$  with two certain and more precise decision rules as follows:

- $R_a$ : with the same structure and type as rule  $R_i$  but the RHS of conditions and the decision are those of rule  $R_j$ ;
- $R_b$ : with the same structure and type as rule  $R_j$  but the RHS of conditions and the decision are those of rule  $R_i$ ;

We mention that we may identify a third situation concerning the assignment of objects where the RHS of the different conditions are in the range  $[r_\alpha, r_\beta]$  (see Figure 3). In this case, there is a contradiction with the initial decision:

- by  $R_i$ , objects with RHS in  $[r_\alpha, r_\beta]$  should be assigned to  $CI_k^<$ , and
- by  $R_j$ , objects with RHS in  $[r_\alpha, r_\beta]$  should be assigned to  $CI_t^>$ ,

Since  $k < t$ , we should assign objects either to  $CI_k^<$  or  $CI_t^>$ . However, to avoid conflict assignments, we opted out not to include an additional rule as in the previous case.

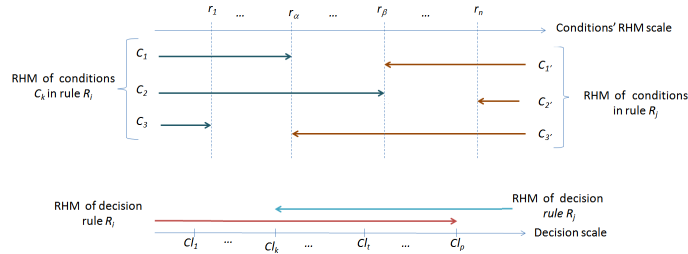


**Fig. 3.** Schematic representation of conditions overlap

**Case 4.  $R_i$  and  $R_j$  have overlapped decisions** This situation is graphically illustrated by Figure 4. In this figure, we assumed that all conditions attributes have the same scale and that  $1 < \alpha < \beta < n$ . As shown in Figure 4 the constraints defined by the conditions of rules  $R_i$  and  $R_j$  are totally disjoint while decisions overlap.

To reduce the interval-based assignments of decision objects, we propose to replace rules  $R_i$  and  $R_j$  with two certain and more precise decision rules as follows:

- $R_a$ : like rule  $R_i$  but the decision is as rule  $R_j$ ;
- $R_b$ : like rule  $R_j$  but the decision is as rule  $R_i$ ;



**Fig. 4.** Schematic representation of decisions overlap

#### 4.2 Step 2: Elimination of Redundant Decision Rules

The objective of this step is to eliminate (i) redundant decision rules; and (ii) rules fully included in other rules. In the second case, two options are possible: either we remove the more general rule or the less general rule. Both solutions may lead to preferential information loss. To minimize the loss of preferential information, we can rely on some measures. Let  $R_a$  and  $R_b$  be two redundant decision rules. Let  $[[R_a]]$  and  $[[R_b]]$  be the sets of decision objects supporting decision rules  $R_a$  and  $R_b$ , respectively. Then, we define the following two measures:

- Information loss:

$$IL(R_a, R_b) = \begin{cases} 0, & \text{if } R_a \subseteq R_b, \\ \frac{[[R_b]] \setminus [[R_a]]}{[[R_b]]}, & \text{otherwise.} \end{cases} \quad (2)$$

$IL(R_a, R_b)$  measures the information loss when decision rule  $R_b$  is removed.

- Precision loss:

$$PL(R_a, R_b) = 1 - IL(R_a, R_b). \quad (3)$$

These two measures vary in different directions and can be used to decide which of decision rules  $R_a$  and  $R_b$  should be removed. It consists of a tradeoff between information loss and precision loss.

## 5 Application

To partially illustrate the proposed algorithm, we consider a real-world data relative to a business school admission where two decision makers (designed by DM1 and DM2 in the rest of the paper) are involved. The learning set is composed of 175 objects (students in this case). A randomly selected extract from the learning set is given in Table 1. In this table, the decisions 'A' and 'R' stand for 'accepted' and 'rejected', respectively. The comparison of the individual assignments shows that the decision makers disagree on 40 (22.86%) students.

**Table 1.** An extract from the learning set.

ID ID	French (fr)	Logic (lo)	General Culture	Collective interview (co)	Face to Face Meeting (ff)	Individual interview (in)	English (en)	Decision DM1 (dec1)	Decision DM2 (dec2)
135	16.8	10.6	6.6	10	yes	13.5	8	A	A
162	9.6	10.6	6.6	14.83	yes	9.5	6.5	R	R
161	14.4	8	10.8	11	yes	5.6	15	R	A
73	12.8	13.4	6.6	14.33	yes	14.8	12.5	A	A
64	13.6	20	10.8	9.83	yes	15.6	10	A	A
110	14.4	10.6	3.4	9	yes	14.9	14	A	R
159	14.4	13.4	9.2	6.83	yes	11.8	6	R	A
89	9.6	10.6	5.8	9.83	yes	18	11.5	A	R
15	12.8	9.4	11.6	?	no	18	15	A	A
1	16.8	12	10.8	18	yes	17.4	19	A	A
117	15.2	9.4	11.6	11.33	yes	12.9	10	A	A
53	10.4	9.4	5.8	14	yes	17.4	14	A	R
144	9.6	12.6	15	?	no	11.3	7.25	A	A
45	13.6	17.4	6.6	18	yes	13.5	12.5	A	A
80	13.6	14.6	5.8	18	yes	14.1	8	A	A
41	13.6	16	10.8	15.33	yes	12.7	16	A	A
147	10	10.6	8.4	?	no	13.5	6	A	R
160	14.4	8	7.6	17	yes	8.5	6	R	A
84	16.8	16	2.6	9.33	yes	15.6	12	A	A
123	17.6	12	5	11	yes	12.5	11	A	A
33	12.8	13.4	9.2	13.67	yes	18	12	A	A
82	14.4	10.6	7.6	18	yes	14.7	7	A	A
14	15.2	15	13.4	?	no	16.6	14	A	A
168	13.6	10.6	5	4	yes	12.7	6	R	R
104	15.2	14.6	13.4	13	yes	10.7	11.5	A	A

We then applied the DRSA two times to approximate this learning set using the assignments given by DM1 and DM2. The application of rule induction algorithm DOMLEM on the obtained approximations leads to two collections of decision rules, which are given Table 2 (for DM1) and Table 3 (for DM2).

**Table 2.** Decision rules relative to DM1.

#	Rule description
1.1	If ( $in \leq 7.5$ ) then ( $dec1 = R$ )
1.2	If ( $en \leq 1.5$ ) then ( $dec1 = R$ )
1.3	If ( $en \leq 4$ ) & ( $gc \leq 5$ ) then ( $dec1 = R$ )
1.4	If ( $en \leq 6.5$ ) & ( $in \leq 11.8$ ) then ( $dec1 = R$ )
1.5	If ( $lo \leq 6.6$ ) & ( $gc \leq 4.2$ ) then ( $dec1 = R$ )
1.6	If ( $in \leq 9$ ) & ( $co \leq 6$ ) then ( $dec1 = R$ )
1.7	If ( $gc \leq 5$ ) & ( $en \leq 6$ ) & ( $in \leq 12.7$ ) then ( $dec1 = R$ )
1.8	If ( $in \geq 15.67$ ) then ( $dec1 = A$ )
1.9	If ( $lo \geq 20$ ) then ( $dec1 = A$ )
1.10	If ( $co \geq 15.6$ ) & ( $lo \geq 17.4$ ) then ( $dec1 = A$ )
1.11	If ( $en \geq 8.5$ ) & ( $co \geq 10.83$ ) & ( $lo \geq 14$ ) then ( $dec1 = A$ )
1.12	If ( $gc \geq 12$ ) then ( $dec1 = A$ )
1.13	If ( $fr \geq 16.8$ ) then ( $dec1 = A$ )
1.14	If ( $in \geq 9.2$ ) & ( $en \geq 4.56$ ) & ( $ff \geq 1$ ) & ( $co \geq 16$ ) then ( $dec1 = A$ )
1.15	If ( $in \geq 11.6$ ) & ( $lo \geq 7.4$ ) & ( $en \geq 4.56$ ) then ( $dec1 = A$ )
1.16	If ( $en \geq 10.75$ ) & ( $gc \geq 8.6$ ) & ( $ff \geq 1$ ) & ( $co \geq 10$ ) & ( $en \geq 12$ ) & ( $in \geq 9.2$ ) then ( $dec1 = A$ )
1.17	If ( $lo \geq 13.4$ ) & ( $en \geq 10.75$ ) & ( $fr \geq 13.6$ ) & ( $gc \geq 7.6$ ) & ( $in \geq 9.83$ ) then ( $dec1 = A$ )

**Table 3.** Decision rules relative to DM2.

#	Rule description
2.1	If (fr $\leq$ 6.6) then (dec2 =R)
2.2	If (lo $\leq$ 5.4) & (gc $\leq$ 8.4) then (dec2 =R)
2.3	If (fr $\leq$ 10.8) & (gc $\leq$ 8.4) & (lo $\leq$ 14.6) then (dec2 =R)
2.4	If (lo $\leq$ 8.0) & (en $\leq$ 6.0) & (in $\leq$ 4.0) then (dec2 =R)
2.5	If (gc $\leq$ 5.0) & (lo $\leq$ 10.6) & (fr $\leq$ 14.4) then (dec2 =R)
2.6	If (lo $\leq$ 8.0) & (en $\leq$ 5.5) then (dec2 =R)
2.7	If (gc $\leq$ 5.8) & (en $\leq$ 9.5) & (lo $\leq$ 12.0) & (in $\leq$ 14.8) & (fr $\leq$ 12.8) then (dec2 =R)
2.8	If (lo $\leq$ 9.4) & (gc $\leq$ 6.6) & (en $\leq$ 11.0) & (in $\leq$ 13.1) then (dec2 =R)
2.9	If (lo $\leq$ 8.0) & (gc $\leq$ 5.8) & (en $\leq$ 9.5) & (fr $\leq$ 16.0) then (dec2 =R)
2.10	If (gc $\leq$ 5.8) & (lo $\leq$ 10.6) & (fr $\leq$ 12.8) then (dec2 =R)
2.11	If (gc $\geq$ 9.6) then (dec2 = A)
2.12	If (lo $\geq$ 15.2) & (en $\geq$ 16.5) then (dec2 = A)
2.13	If (fr $\geq$ 14.4) & (facetoface $\geq$ 1) & (in $\geq$ 18.0) then (dec2 = A)
2.14	If (lo $\geq$ 13.4) & (fr $\geq$ 15.2) then (dec2 = A)
2.15	If (fr $\geq$ 12.0) & (ff $\geq$ 1) & (gc $\geq$ 5.8) & (lo $\geq$ 6.6) & (in $\geq$ 8.0) then (dec2 = A)
2.16	If (fr $\geq$ 17.6) then (dec2 = A)
2.17	If (lo $\geq$ 10.6) & (in $\geq$ 18.0) & (gc $\geq$ 5.8) then (dec2 = A)
2.18	If (lo $\geq$ 17.4) then (dec2 = A)
2.19	If (en $\geq$ 14.0) & (fr $\geq$ 13.6) & (in $\geq$ 16.2) then (dec2 = A)
2.20	If (lo $\geq$ 13.4) & (in $\geq$ 11.67) & (ff $\geq$ 1) & (in $\geq$ 15.67) then (dec2 = A)
2.21	If (en $\geq$ 14.5) & (gc $\geq$ 8.2) then (dec2 = A)
2.22	If (lo $\geq$ 16.0) & (in $\geq$ 16.4) then (dec2 = A)
2.23	If (gc $\geq$ 9.2) & (lo $\geq$ 13.4) then (dec2 = A)

This illustrative example uses only two decision classes. Accordingly, there is no overlap between decision rules of different types. Then, only the second step will be applied to aggregate the decision rules. A careful examination of Table 2 and Table 3 shows that there are three cases of redundancy: (i) Rule 1.9 and Rule 2.18; (ii) Rule 1.12 and Rule 2.11; and (iii) Rule 1.13 and Rule 2.16. The result of the application of Equations (1) and (2) on these pairs of decision rules is summarized in Table 4. Based on these results and to reduce information loss, decision rules 1.18, 2.11 and 1.13 should be removed.

**Table 4.** Information and precision loss.

Case	Rule to remove	IL	PL
1	1.9	0	1
	1.18	0.923	0.077
2	1.12	0	1
	2.11	0.633	0.367
3	1.13	0.245	0.755
	2.16	0	1

We then applied the remaining decision rules to classify all the students. Results show that the obtained collective assignments match with the initial assignments of DM1 for about 96.2% of students and with the initial assignments of DM2 for about 92.3% of students. Thus, DM1 and DM2 need to discuss only a very limited number of conflicting situations (instead on 40 conflicting situations initially).

## 6 Conclusion

We proposed an output oriented aggregation strategy to coherently combine different sets of decision rules obtained from different decision makers. The proposed aggregation algorithm is illustrated by using real-world data relative to business school admission. An important aspect of the proposed approach is that

the consensus between decision makers [13] is computed using objective preference information. In the future, we intend first to apply the proposed aggregation algorithms to other datasets, especially those non-binary decision attributes and with more decision makers. We also intend to study the behavior of the aggregation algorithm with large datasets. Finally, we will intend to design new measures to evaluate information loss, precision loss and information redundancy.

## References

1. Bi, W.J., Chen, X.H.: An extended dominance-based rough set approach to group decision. In: Guizani, M., Chen, H.H., Zhang, X. (eds.) Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing (WiCom 2007). pp. 5753–5756. Shanghai, China (21-25 Sept 2007)
2. Blaszczynski, J., Greco, S., Slowinski, R.: Multi-criteria classification - a new scheme for application of dominance-based decision rules. *European Journal of Operational Research* **181**(3), 1030–1044 (2007)
3. Chakhar, S., Ishizaka, A., Labib, A., Saad, I.: Dominance-based rough set approach for group decisions. *European Journal of Operational Research* **251**(1), 206–224 (2016)
4. Chakhar, S., Saad, I.: Dominance-based rough set approach for groups in multicriteria classification. *Decision Support Systems* **54**(1), 372–380 (2012)
5. Chen, Y., Hipel, K., Kilgour, D.: A decision rule aggregation approach to multiple criteria group decision support. In: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC 2008). pp. 2514–2518. Institute of Electrical and Electronics Engineers (IEEE), Singapore (12-15 Oct 2008)
6. Chen, Y., Kilgour, D., Hipel, K.: A decision rule aggregation approach to multiple criteria-multiple participant sorting. *Group Decision and Negotiation* **21**, 727–745 (2012)
7. Greco, S., Matarazzo, B., Slowinski, R.: Rough sets theory for multicriteria decision analysis. *European Journal of Operational Research* **129**(1), 1–47 (2001)
8. Greco, S., Matarazzo, B., Slowinski, R.: Dominance-based rough set approach to decision involving multiple decision makers. In: Greco, S., Hata, Y., Hirano, S., Inuiguchi, M., Miyamoto, S., Nguyen, H., Slowinski, R. (eds.) Proceedings of the 5th International conference Rough sets and current trends in computing (RSCTC), Kobe, Japan, November 6-8, LNAI, vol. 4259, pp. 306–317. Springer-Verlag, Berlin Heidelberg (2006)
9. Greco, S., Matarazzo, B., Slowinski, R., Stefanowski, J.: An algorithm for induction of decision rules consistent with the dominance principle. In: Ziarko, W., Yao, Y. (eds.) *Rough Sets and Current Trends in Computing*. pp. 304–313. No. 2005 in *Lecture Notes in Artificial Intelligence*, Springer-Verlag (2001)
10. Pawlak, Z.: *Rough set. Theoretical aspects of reasoning about data*. Kluwer Academic Publishers, Dordrecht (1990)
11. Saad, I., Chakhar, S.: Multicriteria methodology based on majority principle for collective identification of company’s valuable knowledge. *Knowledge Management Research & Practice* **10**(4), 380–391 (2012)
12. Xu, Z.: Multiple-attribute group decision making with different formats of preference information on attributes. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on* **37**(6), 1500–1511 (2007)

13. Zhang, H., Dong, Y., Chiclana, F., Yu, S.: Consensus efficiency in group decision making: A comprehensive comparative study and its optimal design. *European Journal of Operational Research* **275**(2), 580–598 (2019)