

A Heuristic Approach to Dynamically Update Approximations of Dominance Based Rough Set Theory

Faryal Nosheen*, Usman Qamar*, Muhammad Summair Raza**

*Department of Computer and Software Engineering, College of Electrical and Mechanical Engineering, National University of Sciences and Technology, Islamabad, Pakistan

**Department of Computer Science and Information Technology, Virtual University of Pakistan, Lahore, Pakistan

faryal.nosheen@ceme.nust.edu.pk, usmanq@ceme.nust.edu.pk, sraza@vu.edu.pk

Abstract— In multi-criteria decision-making problems, dominance based rough set approach (DRSA) has been extensively used. However, in real-time domain, information often evolves with time. To extract updated knowledge from dynamic systems using DRSA, we have to re-calculate approximations to update information. This process of periodic re-calculation consumes a significant computational cost, and it considerably affects the efficiency of conventional algorithm. Therefore, to address this challenge, in this paper, we focused on updating approximation sets of DRSA, while considering the evolution in the object set of the system. We propose an algorithm which avoid much redundant computations. The experimental evaluation of proposed approach on UCI publicly available data sets, shows that the proposed technique, surpasses the conventional approach of DRSA in terms of execution time. We achieve same approximation sets but computational time decreases by almost 98%. This shows that our method can be used successfully for dynamic applications based on DRSA approximations.

Keywords— Decision System, Rough Set Theory, Dominance Based Rough Set Theory, Dynamic update, Dynamic System.

I. INTRODUCTION

Pawlak [1] proposed an idea of using rough sets for data mining in 1982, named as Rough Set Theory (RST). RST is one of the mathematical based knowledge induction techniques which can be used for information discovery from relational databases. But we can have a problem where relations in a data hold preference order. To address this situation in the domain of rough sets, Greco et al. [2-4] presented the concept of dominance relation instead of equivalence relation and called it dominance-based rough sets theory-DRSA. It has been widely applied in many applications i.e., customer behavior analysis, risk-predictions, business evaluations, airline services prediction, attribute reduction [5-10]. The traditional algorithm of DRSA and its many flavors, deal with the static information system. However, in real-time systems, information often experiences changes over time. It raises an issue of dynamic maintenance of system to get updated information while making data evaluations, predictions, information analysis, classification etc. To explore different, researchers proposed

methodologies for three types of variations i.e., object set variation [11-13], variation in attribute set [14-16] and variation in values of attributes [17,18]. In this study, we focused on object set variation and proposed an efficient approach to update DRSA approximation sets.

For a single approximation, conventional DRSA algorithm compares every single instance and its characteristics with whole dataset repeatedly [34]. In this paper, we proposed heuristic rules, to update approximation sets efficiently, when changes occur in the object set of the system. These rules have successfully updated all approximations while avoiding unnecessary traversing of the whole dataset for each step. To test the performance of our proposed method, we have conducted experiment on ten UCI datasets [35] and compared performance with conventional algorithm and another incremental method presented in [33]. The results showed that our proposed approach reduced execution time by almost 98%.

The paper is structured as: In section II we present basic concepts of DRSA. In section III related work have been discussed. The proposed approach is described in Section IV. An experimental result presented in Section V. Section VI summarizes our current work and the prospects of future work.

II. DRSA-NOTIONS

A decision system is defined as

$$IS = (U, Q, V, f) \quad (1)$$

Here, universe = U : is a decision table, $(C \cup D) = Q$: it contains conditional attributes and decision attribute., $V = \bigcup_{q \in Q} V_q$: every conditional attribute q has specific value set V_q and $f(x, q)$: this function defines the dominance relation of each object 'x'.

DRSA considers the preference order of the objects by analyzing the dominance relation. We can define dominance relation of objects as follows:

- Dominance positive relation: for mentioned criteria P which is $(P \subseteq C)$, if every value of object x dominates object y

$$D_p^+(x) = \{y \in U; x D_p y\} \quad (2)$$
- Dominance negative relation: if value of object y for criteria P where $(P \subseteq C)$, is greater than value of object x

$$D_p^-(x) = \{y \in U; y D_p x\} \quad (3)$$

In DRSA decision classes further classified based on preference order as:

- Upward union of classes ($Cl_t^{\geq}$): it includes all objects of class t and higher than class t .

$$Cl_t^{\geq} = U_{s \geq t} Cl_s \quad (4)$$

- Downward union of classes ($Cl_t^{\leq}$): it contains all objects of class t and less preferred than class t

$$Cl_t^{\leq} = U_{s \leq t} Cl_s \quad (5)$$

Another important concept in DRSA is lower and upper approximation. These approximations further classified as

- Lower approximation for upward and downward union of classes respectively

$$\underline{P}(Cl_t^{\geq}) = \{x \in U: D_P^+(x) \subseteq Cl_t^{\geq}\} \quad (6)$$

$$\underline{P}(Cl_t^{\leq}) = \{x \in U: D_P^-(x) \subseteq Cl_t^{\leq}\} \quad (7)$$

- Upper approximation for upward and downward union of classes respectively

$$\overline{P}(Cl_t^{\geq}) = \{x \in U: D_P^+(x) \cap Cl_t^{\geq} \neq \emptyset\} \quad (8)$$

$$\overline{P}(Cl_t^{\leq}) = \{x \in U: D_P^-(x) \cap Cl_t^{\leq} \neq \emptyset\} \quad (9)$$

III. RELATED WORK

To better utilizes the benefits of RST and DRSA for decision making, predictions and behaviour analysis; efficient computation of approximations is much needed. For dynamic applications, conventional DRSA-based algorithms update the information by re-calculating approximations for whole decision system which makes computation time too high.

Many researchers work on different dimensions of dynamic dataset and presented solutions to update approximations [19-20]. When variation occur in a single object, Jia et al. [21] proposed an approach to update rules using the dominance based rough set theory. Similarly, Li et al. [22] presented an approach to update lower and upper approximations when an object enters or leaves the system, using P-generalized decisions [23]. In this method, whole process of calculation of dominance relation needs to be repeated for all new objects. In [24], authors discussed the approximations updating methods when an object enters or leave the system in a multi-criteria classification domain. Luo et. al., [25-27] proposed three different algorithms to update algorithms for changes in object set, attribute set, and criteria values. Wang et al. presented an approach to handle the changes in object set and attribute set simultaneously while updating approximations [28]. In [29], Chen et al. proposed an algorithm to efficiently update approximation sets of DRSA for incomplete information system upon occurrence of variation in the attributes of the objects. Luo et al. proposed an approach to process hierarchical attribute value using DRSA and introduced an algorithm to update approximations in case of dynamic data [30]. Sarra et al. [31] proposed an algorithm using P-generalized decisions [23] for updating approximations of DRSA.

Li et.al [32] proposed solution for dynamain maintenance of approximations in case of multiple objects change with time, authors computed dominance relation for new objects and then update approximations. In [33] Li et.al. proposed a methodology of considering new objects as new datasets. Authors suggested to repeat whole conventional process of

calculation of approximations for old and new datasets. However, most of these algorithms have complex structure and repeat various computations against variation of single object.

IV. PROPOSED METHODOLOGY

In this section, we will discuss the methodology to maintain approximation sets of DRSA when an object set evolve with time. We assume that the old values of approximations are known. We propose two different algorithms.

A. An Algorithm to Update Approximations Upon Insertion of New Object- INO

We assume that a set of new objects, U^+ , becomes a part of an information system at some time $t + 1$. In Table 1, we mention all symbols used in this paper. First, we update decision classes, pseudocode is given in Figure. 1.

Table 1. DETAIL OF SYMBOLS

Description	Notions of DRSA at time t	Notions of DRSA at time $t+1$
Decision System	U	$\hat{U}$
Decision class t	Cl_t	Cl'_t
More preferred decision class than 't'	$Cl_t^{\geq}$	$Cl'^{\geq}_t$
Less preferred decision class than 't'	$Cl_t^{\leq}$	$Cl'^{\leq}_t$
Lower Approximation for $Cl_t^{\leq}$	$\underline{P}(Cl_t^{\leq})$	$\underline{P}'(Cl'^{\leq}_t)$
Upper Approximation for $Cl_t^{\leq}$	$\overline{P}(Cl_t^{\leq})$	$\overline{P}'(Cl'^{\leq}_t)$
Lower Approximation for $Cl_t^{\geq}$	$\underline{P}(Cl_t^{\geq})$	$\underline{P}'(Cl'^{\geq}_t)$
Upper Approximation for $Cl_t^{\geq}$	$\overline{P}(Cl_t^{\geq})$	$\overline{P}'(Cl'^{\geq}_t)$

Input: U^+ , $\hat{U} = U \cup U^+$, $d = \text{total number of classes}$

Output: $Cl'_t, Cl'^{\geq}_t, Cl'^{\leq}_t$

For $t=1$ to d // d holds total number of decision class i.e. 2, 3

For $i=1$ to $|\hat{U}|$

If $(x_i, d) \in \text{class } t$

$Cl'_t = Cl_t \cup \{x_i\}$

Else if $(x_i, d) \in \text{more preferred class than class } t$

$Cl'^{\geq}_t = Cl_t^{\geq} \cup \{x_i\}$

Else if $(x_i, d) \in \text{less preferred class than class } t$

$Cl'^{\leq}_t = Cl_t^{\leq} \cup \{x_i\}$

End if

End for

End for

Figure 1. Pseudocode to update decision classes

In the second step, we give the definition of heuristic rules to update lower and upper approximations. Using these rules when we compare single object, our algorithm selects it for all relevant approximations sets simultaneously.

1) Heuristic Rules:

Proposition 1

- Let $x^+ \in U^+$, is a new object of decision class t :

$$\underline{P}'(Cl'^{\leq}_t) = \begin{cases} \underline{P}(Cl_t^{\leq}) \cup \{x^+ \in U^+: \text{if } x^+ \text{ is less preferred than all objects of } Cl_t^{\leq}\} & \text{if } t < 2 \\ \underline{P}(Cl_t^{\leq}) \cup \{x^+ \in U^+: \text{if } x^+ \text{ is less preferred than all objects of } Cl_t^{\leq}\} \cup \{x^+ \in Cl_t^{\leq}: \text{if } x^+ \text{ is more preferred than object of } Cl'_t\} \cup \underline{P}'(Cl'^{\leq}_{t-1}) & \text{if } t \geq 2 \end{cases}$$

x^+ selects for $\underline{P}'(Cl'^{\leq}_t)$, if it meets following conditions:

- $x^+ \in Cl'_t$ and less preferred than all objects of $Cl'^{\geq}_t$

- ii. $x^+ \in Cl'_t$ and more preferred than all objects of Cl'_t ($t \geq 2$)
- iii. If $t \geq 2$, all objects of $\underline{P}'(Cl'_{t-1})$ become part of $\underline{P}'(Cl'_t)$.

Proposition 2

Let $x^+ \in U^+$, is new object of decision class t .

$$\overline{P}'(Cl'_t) = \begin{cases} \overline{P}(Cl'_t) \cup \{x^+ \in U^+\} \cup \{x^+ \in U^+, y \in Cl'_t : y \text{ is less preferred than } x^+\} & \text{if } t < 2 \\ \overline{P}(Cl'_t) \cup \{x^+ \in U^+ \text{ or } Cl'_t\} \cup \{(x^+ \in U^+ y \in Cl'_t : y \text{ is less preferred than } x^+) \wedge (y \cap \overline{P}(Cl'_t)) = \emptyset\} & \text{if } t \geq 2 \end{cases}$$

A new object x^+ must have the following properties

- i. $x^+ \in Cl'_t$ and/or $x^+ \in Cl'_t$, when ($t \geq 2$)
- iii. $x^+ \in Cl'_t$ and **more preferred than**, object y where $y \in Cl'_t$ and object y does not exist in $\overline{P}(Cl'_t)$, then object y selects for $\overline{P}'(Cl'_t)$.

Proposition 3

Let $x^+ \in U^+$, is an added new object of decision class t where $t \geq 2$.

$$\underline{P}'(Cl'_t) = \underline{P}(Cl'_t) \cup \{x^+ \in U^+, (x^+ \cap \overline{P}(Cl'_{t-1})) = \emptyset\} \quad \text{if } t \geq 2$$

As per defined proposition, if x^+ selects for $\overline{P}'(Cl'_{t-1})$, it does not belong to $\underline{P}'(Cl'_t)$ otherwise it belongs to $\underline{P}'(Cl'_t)$.

Proposition 4

Let $x^+ \in U^+$, is an added new object of decision class t .

$$\overline{P}'(Cl'_t) = \overline{P}(Cl'_t) \cup \{x^+ \in U^+ \text{ or } Cl'_t\} \cup \{(x^+ \in Cl'_t, y \in Cl'_t : x^+ \text{ more preferred than object } y) \wedge (x^+ \cap \overline{P}(Cl'_t)) = \emptyset\} \quad \text{if } t \geq 2$$

x^+ must fulfil the following conditions to select for $\overline{P}'(Cl'_t)$

- i. If $x^+ \in U^+$ and $x^+ \in Cl'_t$, when $t \geq 2$
- iii. If $x^+ \in Cl'_t$, **more preferred than**, y where $y \in Cl'_t$ and x^+ does not already exists in $\overline{P}(Cl'_t)$, then x^+ selects.

2) Pseudocode:

Input: $\underline{P}(Cl'_t), \overline{P}(Cl'_t), \underline{P}(Cl'_t), \overline{P}(Cl'_t), Cl'_t, Cl'_t, Cl'_t, U^+$
Output: $\underline{P}'(Cl'_t), \overline{P}'(Cl'_t), \underline{P}'(Cl'_t) \& \overline{P}'(Cl'_t)$
 For $i=1$ to U^+ , row count
 if $(x_i^+.D) < 2$
 $t = x_i^+.D$ // D symbolize decision class
 $\overline{P}'(Cl'_t) = \overline{P}(Cl'_t) \cup \{x_i^+\}$
 $\overline{P}'(Cl'_{t+1}) = \overline{P}(Cl'_{t+1}) \cup \{x_i^+\}$
 greater = false // a boolean variable
 For $j=1$ to Cl'_t // Compare x_i^+ with x_j
 if x_i^+ more preferred or equal to x_j
 greater = true
 $m = x_j.D$
 $\overline{P}'(Cl'_t) = \overline{P}(Cl'_t) \cup \{x_j\}$
 $\underline{P}'(Cl'_m) = \underline{P}(Cl'_m) \cup \{x_i^+\}$
 $\overline{P}'(Cl'_m) = \overline{P}(Cl'_m) \cup \{x_i^+\}$
 End if
 End for
 if greater = false
 $\underline{P}'(Cl'_t) = \underline{P}(Cl'_t) \cup \{x_i^+\}$
 End if
 End if
 greater = false
 if $(x_i^+.D) \geq 2$
 $t = x_i^+.D$ // assign value of decision class to t
 $\overline{P}'(Cl'_t) = \overline{P}(Cl'_t) \cup \{x_i^+\}$
 $\overline{P}'(Cl'_t) = \overline{P}(Cl'_t) \cup \{x_i^+\}$
 If ($t > 2$)
 For $m = 1$ to $t-1$
 $\overline{P}'(Cl'_{t-m}) = \overline{P}(Cl'_{t-m}) \cup \{x_i^+\}$
 End for

End if
 if $x_i^+ \notin \overline{P}'(Cl'_{t-1})$
 $\underline{P}'(Cl'_t) = \underline{P}(Cl'_t) \cup \{x_i^+\}$
 End if
 For $j=1$ to Cl'_t // Compare x_i^+ with x_j
 if x_i^+ more preferred or equal to x_j
 greater = true
 $\overline{P}'(Cl'_t) = \overline{P}(Cl'_t) \cup \{x_j\}$
 End if
 End for
 if greater = false
 $\underline{P}'(Cl'_t) = \underline{P}(Cl'_t) \cup \{x_i^+\}$
 End if
 End if
 End for

Figure 2. Pseudocode of Proposed Algorithm-INO

B. An Algorithm to Update Approximations Upon Deletion of Old Object- DOO

Here we discuss the scenario when set of objects U^- leaves the system at time $t + 1$. Below we defined the rules to update approximations (symbols defined in Table-1).

3) Heuristic Rules:**Proposition 1**

Let $x^- \in U^-$, is an object which leaves the system

$$\underline{P}'(Cl'_t) = \underline{P}(Cl'_t) - \{x^-\} \text{ if } x^- \in \underline{P}(Cl'_t)$$

$$\underline{P}'(Cl'_{t+1}) = \underline{P}(Cl'_{t+1}) - \{x^-\} \text{ if } x^- \in \underline{P}(Cl'_t)$$

As defined in above proposition

- i. An object $x^- \in$ class t , we check if $x^- \in \underline{P}(Cl'_t)$, remove x^- from $\underline{P}(Cl'_t)$ and $\underline{P}(Cl'_{t+1})$

Proposition 2

Let $x^- \in U^-$, leaves the system.

- i. An object $x^- \in$ class t , remove x^- from $\overline{P}(Cl'_t)$ and $\overline{P}(Cl'_{t+1})$.
- ii. An object $x^- \in$ class t , $y \in Cl'_t$: if x^- is **more preferred than** y , y will be deleted from $\overline{P}(Cl'_t)$ set.
 $\overline{P}'(Cl'_{t+1}) = \overline{P}(Cl'_{t+1}) - \{x^-\} \quad \text{if } t \leq 2$

$$\overline{P}'(Cl'_t) = \begin{cases} \overline{P}(Cl'_t) - \{x^-\} \\ \overline{P}(Cl'_t) - \{y\}, & \text{if } x^- \in Cl'_t, y \in Cl'_t : \text{if } x^- \text{ is} \\ \text{more preferred than } y \end{cases}$$

Proposition 3

Let $x^- \in U^-$, is an object leave the system

$$\underline{P}'(Cl'_t) = \underline{P}(Cl'_t) - \{x^-\} \text{ if } (x^- \in Cl'_t, y \in Cl'_t : \text{if } x^- \text{ is more preferred than } y) \wedge (t \geq 2)$$

An object x^- affects the following approximation sets

- i. An object $x^- \in U^-$, object $y \in Cl'_t$ if x^- is **more preferred than** y and decision class of x^- i.e., t , is more preferred than class of object y , we will remove x^- from $\underline{P}(Cl'_t)$

Proposition 4

Let $x^- \in U^-$, is an object, check following conditions:

- i. An object $x^- \in Cl'_t$, we remove it from $\overline{P}(Cl'_t)$ and if $t \geq 2$, remove x^- from $\overline{P}(Cl'_{t+1})$.

- ii. An object $x^- \in Cl_t^l$, an object $y \in Cl_t^s$, if x^- is *less preferred or equal to* y and $t \geq 2$, delete object y from $\bar{P}(Cl_t^s)$ and $\bar{P}(Cl_{t+1}^s)$.

$$\bar{P}(Cl_t^s) = \begin{cases} \bar{P}(Cl_t^s) - \{x^-\} & \text{if } t \geq 2 \\ \bar{P}(Cl_t^s) - \{y\} & \text{if } \{x^- \in U^-, y \in Cl_t^s : \text{if } x^- \text{ is less preferred or equal to } y\} \wedge (t \geq 2) \end{cases}$$

$$\underline{P}(Cl_t^s) = \underline{P}(Cl_t^s) - \{y\} \quad \text{if } \{x^- \in U^-, y \in Cl_t^s : \text{if } x^- \text{ is less preferred or equal to } y\} \wedge (t \geq 2)$$

$$\bar{P}(Cl_{t+1}^s) = \bar{P}(Cl_{t+1}^s) - \{x^-\} \quad \text{if } t \geq 2$$

$$\bar{P}(Cl_{t+1}^s) = \bar{P}(Cl_{t+1}^s) - \{x^-\} \quad \text{if } \{x^- \in U^-, y \in Cl_{t+1}^s : \text{if } x^- \text{ is less preferred or equal to } y\} \wedge (t \geq 2)$$

2) Pseudocode:

Input: $Cl_t^l, Cl_t^s, Cl_t^l, \underline{P}(Cl_t^s), \bar{P}(Cl_t^s), \underline{P}(Cl_t^s)$ and $\bar{P}(Cl_t^s), U^-$
Output: $\underline{P}(Cl_t^s), \bar{P}(Cl_t^s), \underline{P}(Cl_t^s), \bar{P}(Cl_t^s)$

row = U^- . row count
 For i=1 to row
 $\bar{U} = U - \{x_i^-\}$
 if $(x_i^-, d) \in$ class t // update decision classes
 $Cl_t^l = Cl_t^l - \{x_i^-\}$
 Else if $(x_i^-, d) \in$ *more preferred* class than class t
 $Cl_t^s = Cl_t^s - \{x_i^-\}$
 Else if $(x_i^-, d) \in$ *less preferred* class than class t
 $Cl_t^s = Cl_t^s - \{x_i^-\}$
 // update approximation sets upon deletion of each object
 if $(x_i^-, d) < 2$
 $\bar{P}(Cl_t^s) = \bar{P}(Cl_t^s) - \{x_i^-\}$
 $\bar{P}(Cl_{t+1}^s) = \bar{P}(Cl_{t+1}^s) - \{x_i^-\}$
 $t = x_i^-, d$
 For j=1 to Cl_t^s
 If x_i^- more preferred than x_j
 $m = x_j, d$
 $\bar{P}(Cl_t^s) = \bar{P}(Cl_t^s) - \{x_j\}$
 $\underline{P}(Cl_m^s) = \underline{P}(Cl_m^s) - \{x_i^-\}$
 End if
 End for
 For n=t to d
 $\bar{P}(Cl_n^s) = \bar{P}(Cl_n^s) - \{x_i^-\}$
 If $x_i^- \in \bar{P}(Cl_{n+1}^s)$
 $\bar{P}(Cl_{n+1}^s) = \bar{P}(Cl_{n+1}^s) - \{x_i^-\}$ // lowest class is 1
 End if
 if $x_i^- \in \underline{P}(Cl_t^s)$
 $\underline{P}(Cl_t^s) = \underline{P}(Cl_t^s) - \{x_i^-\}$
 $\underline{P}(Cl_{n+1}^s) = \underline{P}(Cl_{n+1}^s) - \{x_i^-\}$ // lowest class is 1
 End if
 End for
 End if
 if $(x_i^-, d) \geq 2$
 $t = x_i^-, d$
 if $x_i^- \in \underline{P}(Cl_t^s)$
 $\underline{P}(Cl_t^s) = \underline{P}(Cl_t^s) - \{x_i^-\}$
 End if
 $\bar{P}(Cl_t^s) = \bar{P}(Cl_t^s) - \{x_i^-\}$
 $\bar{P}(Cl_t^s) = \bar{P}(Cl_t^s) - \{x_i^-\}$
 If $(t < d)$
 For m=t to d
 If $x_i^- \in \bar{P}(Cl_{m+1}^s)$
 $\bar{P}(Cl_{m+1}^s) = \bar{P}(Cl_{m+1}^s) - \{x_i^-\}$
 End if
 End for
 End if
 if $(t = d)$
 $\underline{P}(Cl_t^s) = \underline{P}(Cl_t^s) - \{x_i^-\}$
 End if

m=0
 For j=1 to Cl_t^s // Compare x_i^- with objects of Cl_t^s
 if x_i^- less preferred or equal to x_j
 $m = x_j, d$
 $\underline{P}(Cl_t^s) = \underline{P}(Cl_t^s) - \{x_j\}$
 $\bar{P}(Cl_t^s) = \bar{P}(Cl_t^s) - \{x_j\}$
 $\bar{P}(Cl_m^s) = \bar{P}(Cl_m^s) - \{x_i^-\}$
 Else $\underline{P}(Cl_t^s) = \underline{P}(Cl_t^s) - \{x_i^-\}$
 End if
 End for
 if $(t < d)$
 For j=1 to Cl_t^s // Compare x_i^- with objects of Cl_t^s
 $m = x_j, d$
 if x_i^- more preferred than x_j
 $\bar{P}(Cl_{t+1}^s) = \bar{P}(Cl_{t+1}^s) - \{x_j\}$
 $\bar{P}(Cl_t^s) = \bar{P}(Cl_t^s) - \{x_j\}$
 End if
 End for
 End if
 End if
 End for

Figure 3. Pseudocode of Proposed Algorithm- DOO

C. Illustrative Example

We consider the sample information system given below:

TABLE 2. SAMPLE DECISION SYSTEM

Object	q1	q2	q3	d
x_1	2	2	1	1
x_2	3	2	1	2
x_3	2	3	1	2
x_4	1	2	3	1
x_5	1	1	2	1
x_6	1	2	2	2
x_7	3	3	1	3
x_8	3	2	2	2
x_9	2	2	3	3
x_{10}	3	2	3	3

Assume that we have previous information about the decision classes and approximations.

1) Update Approximations Upon Insertion of New Objects:

To clarify the implementation of our proposed approach, we supposed that a new set of objects $U^+ = \{x_{11}, x_{12}, x_{13}, x_{14}\}$ entered in decision system, detail of conditional and decision attribute is given as $(x_i, q1, q2, q3, d)$:

- $(x_{11}, 1, 2, 3, 1), (x_{12}, 2, 2, 3, 3), (x_{13}, 2, 1, 2, 2), (x_{14}, 1, 2, 2, 2)$

Let's consider new object x_{11} . After updating decision classes and respective higher/lower preference ordered classes, we update the approximation sets.

- As per proposition 1, decision class of x_{11} is lowest class, so it selects for $\bar{P}(Cl_1^s)$. Simultaneously x_{11} selects for $\bar{P}(Cl_2^s)$, as defined in proposition 2 (for $t \geq 2$).
- We compare x_{11} , with objects of Cl_1^s , x_{11} is more preferred than x_6 and x_{14} . So, x_{11} does not select for $\underline{P}(Cl_1^s)$ but as per proposition 1, x_{11} selects for $\underline{P}(Cl_2^s)$. x_{11} also becomes part of $\bar{P}(Cl_2^s)$ (as per proposition 4).
- As mentioned earlier in (ii), objects of Cl_1^s i.e., x_6 and x_{14} are less preferred than x_{11} , so x_{14} selects for updated $\bar{P}(Cl_1^s)$ (x_6 already presents in $\bar{P}(Cl_1^s)$).

iv. As per proposition 2, all objects of $CI_t^<$ select. As x_{11} belongs to class ' $t=1$ ', a less preferred class than class ' $t=2$ ', therefore, we select it for $\bar{P}'(CI_2^<)$.

After completion of these steps, following approximation sets updated: $\bar{P}'(CI_1^<) = \{x_{11}, x_{14}\}$, $\underline{P}'(CI_2^<) = \{x_{11}\}$, $\bar{P}'(CI_2^<) = \{x_{11}\}$, $\bar{P}'(CI_2^>) = \{x_{11}\}$.

Similarly, our algorithm updates lower and upper approximations considering remaining three objects of U^+ .

2) Update Approximations Upon Deletion of Objects: We assume that the following objects $U^- = \{x_3, x_7\}$ leave a decision system at some time $t+1$. When x_3 leaves the system, our method updates decision classes, and approximations, as under:

- To update $\underline{P}'(CI_2^<)$, we use proposition 1 (for $t \geq 2$) as decision class of x_3 ' $t=2$ ' (considering Table I). We check x_3 belongs to $\underline{P}(CI_t^<)$, so, delete it from approximation set.
- We delete x_3 from $\bar{P}(CI_2^<)$ and $\bar{P}(CI_2^>)$, as per proposition 2 & 4 (for $t \geq 2$). Simultaneously, we check whether x_3 belongs to $\bar{P}(CI_{t+1}^<)$ then delete x_3 from $\bar{P}(CI_3^<)$.
- When we compare x_3 with all objects of $CI_2^<$, x_3 appears more preferred than all objects of $CI_2^<$. Therefore, delete it from $\underline{P}(CI_2^<)$, as defined in proposition 3 (for $t \geq 2$).
- We compare x_3 with all objects of $CI_2^>$, as per defined in proposition 2 and find that x_3 is less preferred than all objects of $CI_2^>$, so no action performs here.

Similarly, our algorithm updates approximation sets as per rules in section IV (B) upon deletion of x_7 .

V. RESULTS AND ANALYSIS

In this section we study the efficiency of proposed approach while updating approximations when objects immigrating and emigrating. To test the efficiency of our proposed technique, we have used ten publicly available datasets from UCI library [35] i.e., musk2, chess, EEG eye state, breast cancer wisconsin, abalone, glass, wine, iris, sonar, car evaluation. These datasets have numerical values for conditional attributes and decision classes. To extensively study the performance of our proposed algorithms, we do comparison with conventional DRSA algorithm and another incremental algorithm which worked in dynamic environment.

We compared conventional and proposed algorithms with variation ratio of 5% of original dataset size. We selected random values for conditional and decision attributes of new objects, with an assumption that value set remains same. We have old approximation sets and we updated DRSA approximations using proposed algorithm-INO and conventional DRSA algorithm. We compared the computational time of both algorithms by calculating the percentage decrease in execution time. Our proposed algorithm-INO updated approximations in reduce time as compared to conventional, with an average reduction of 98% (Table 3)

TABLE 3. COMPUTATION TIME UPON INSERTION OF NEW OBJECTS

Dataset	Time Taken (s)		Percentage reduction in time
	Conventional Algorithm	Proposed Algorithm-INO	
Musk2	66.93	0.915	98.63%

Chess	54.75	0.7	98.72.%
EEG Eye State	60.13	0.53	99.11%
Breast Cancer	5.1	0.06	98.12%
Abolane	138.22	3.47	97.48%
Glass	0.2995	0.005	98.33%
Wine	0.1535	0.003	97.65%
Iris	0.1319	0.007	94.61%
Sonar	0.5999	0.006	98.91%
Car Evaluation	12.76	0.2	98.43%

Similarly, we compared execution time of conventional algorithm and proposed algorithm-DOO, upon deletion of 5% objects from each dataset and achieved average reduction of 97% as shown in Table 4.

TABLE 4. COMPUTATION TIME UPON DELETION OF OBJECTS

Dataset	Time Taken (s)		Percentage Reduction in Time
	Conventional Algorithm	Proposed Algorithm-DOO	
Musk2	63.93	1.0	98.43%
Chess	52.75	0.791	98.50.%
EEG Eye State	59.23	1.63	97.24%
Breast Cancer	4.77	0.173	96.37%
Abolane	136.62	3.113	97.72%
Glass	0.2835	0.003	98.76%
Wine	0.1395	0.001	98.71%
Iris	0.1279	0.006	95.23%
Sonar	0.5879	0.011	98.12%
Car Evaluation	11.92	0.179	98.49%

Moreover, the computational complexity of our proposed algorithms also reduced to quadratic ($O(U^2)$) as compared to the cubic ($O(U^3)$) complexity of conventional DRSA algorithm. We also compared our proposed algorithm with another incremental algorithm proposed in [33] using 5% variation ratio. In [33], authors considered preference order for conditional attributes only. We implemented both approaches on four large datasets [35]. Results showed that our proposed algorithm performed faster in terms of execution time, as shown in Table 5 and Figure 4.

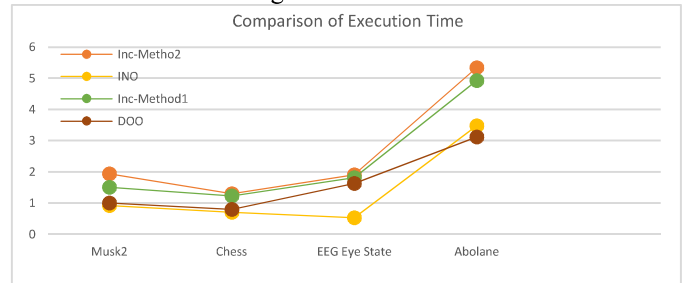


Figure 4. Incremental Method Vs. Proposed Algorithm

TABLE 5. COMPARISON OF INCREMENTAL METHOD & PROPOSED APPROACH

Data set	Time Taken (s)		Percent age Reducti on in Time	Time Taken (s)		Percent age Reducti on in Time
	Incremen tal Method- Insertion	Proposed Appr oach-INO		Incremen tal Method- Deletion	Proposed Appr oach- DOO	
Musk 2	1.93	0.915	98.63%	1.5	1.0	98.43%
Chess	1.3	0.7	98.72.%	1.23	0.79	98.50.%
EEG Eye State	1.9	0.53	99.11%	1.81	1.63	97.24%
Abolane	5.341	3.473	97.48%	4.921	3.11	97.72%

VI. CONCLUSION AND FUTURE WORK

In dynamic data environment, updated knowledge extraction is critical for other dependent tasks i.e., feature selection, behaviour analysis etc. In this research we have studied different mechanisms for updating DRSA approximations when data changes. We have focused on the situation when multiple objects enter and leave the system and proposed rule-based approach, to update approximation sets. Our proposed method avoids the computation of dominance relation for each object which consequently reduce the time require to update approximations. Extensive experiments are conducted on UCI datasets. As results discussed in section V, our proposed algorithm proves efficient as compared to DRSA traditional algorithm. We also compare our approach with another recent incremental method and experimental results have shown the 98% reduction in execution time. In future, we want to explore the variation in the value set of attributes (conditional and decision attribute) along with object set variations.

REFERENCES

- [1] Zdzisław P, Rough sets, *Int. J. Comput. Inf. Sci.* 11 (1982) 341–356.
- [2] Greco S, Matarazzo B, Slowinski R (1999) Rough approximation of a preference relation by dominance relation. *Eur J Oper Res* 117
- [3] Greco S, Matarazzo B, Slowinski R (2001) Rough sets theory for multicriteria decision analysis. *Eur J Oper Res* 129(1):1–47
- [4] Greco S, Matarazzo B, Slowinski R (2002) Rough approximation by dominance relations. *Int J Intell Syst* 17(2):153–171
- [5] Liou, J.: A novel decision rules approach for customer relationship management of the airline market. *Expert Syst. Appl.* 36, 4374–4381 (2009)
- [6] Liou, J., Tzeng, G.: A dominance-based rough set approach to customer behavior in the airline market. *Inf. Sci.* 180, 2230–2238 (2010a)
- [7] Liou, J., Yen, L., Tzeng, G.: Using decision rules to achieve mass customization of airline services. *Eur. J. Oper. Res.* 205, 680–686 (2010b)
- [8] Liou, J., Yen, L., Tzeng, G.: Using decision rules to achieve mass customization of airline services. *Eur. J. Oper. Res.* 205, 680–686 (2010b)
- [9] Błaszczyński J, Greco S, Slowinski R (2009) Monotonic variable consistency rough set approaches. *Int J Approx Reason* 50(7):979–999
- [10] Inuiguchi M, Yoshioka Y, Kusunoki Y. Variable-precision dominance-based rough set approach and attribute reduction. *Int J Approx Reason* 2009;50:1199–1214.
- [11] Shan L, Ziarko W. Data-based acquisition and incremental modification of classification rules. *Comput Intell* 1995;11:357–370.
- [12] Bang W, Bien Z. New incremental learning algorithm in the framework of rough set theory. *Int J Fuzzy Syst* 1999;1:25–36.
- [13] Tong L, An L. Incremental learning of decision rules based on rough set theory. In: *Proc World Congress on Intelligent Control and Automation (WCICA2002)*; 2002. pp 420–425.
- [14] Chan C. A rough set approach to attribute generalization in data mining. *Inform Sci* 1998;107:177–194.
- [15] Li T, Ruan D, Geert W, Song J, Xu Y. A rough sets based characteristic relation approach for dynamic attribute generalization in data mining. *Knowl-Based Syst* 2007;20:485–494.
- [16] Cheng Y. The incremental method for fast computing the rough fuzzy approximations. *Data Knowl Eng* 2011;70:84–100.
- [17] Chen H, Li T, Qiao S, Ruan D. A rough set based dynamic maintenance approach for approximations in coarsening and refining attribute values. *Int J Intell Syst* 2010;25:1005– 1026.
- [18] Chen H, Li T, Ruan D. Maintenance of approximations in incomplete ordered decision systems while attribute values coarsening or refining. *Knowl-Based Syst* 2012;31:140–161.
- [19] H. Chen, T. Li, D. Ruan, J. Lin, C. Hu, A rough-set-based incremental approach for updating approximations under dynamic maintenance environments, *IEEE Trans. Knowledge Data Eng.* 25 (2) (2013) 274–284.
- [20] Y. Cheng, The incremental method for fast computing the rough fuzzy approximations, *Data Knowledge Eng.* 70 (1) (2011) 84–100.
- [21] D. Liu, T. Li, D. Ruan, J. Zhang, Incremental learning optimization on knowledge discovery in dynamic business intelligent systems, *J. Global Optim.* 51 (2) (2011) 325–344.
- [22] Jia X, Shang L, Chen J (2011) Incremental versus non-incremental: data and algorithms based on ordering relations. *Int J Comput Intell Syst* 4:112–122
- [23] Li SY, Li TR, Liu D (2013) Dynamic maintenance of approximations in dominance-based rough set approach under the variation of the object set. *Int J Intell Syst* 28(8):729–751
- [24] Kotłowski, W., Dembczynski, K., Greco, S., Słowiński, R., 2008. Stochastic dominance-based rough set model for ordinal classification. *Information Sciences* 178 (21), 4019–4037.
- [25] Li Y, Liu XQ, Hou JJ (2014) Incrementally updating method in dominance-based rough set approach. *Int J Intell Inf Manag Sci* 3(1):18–22
- [26] Luo C, Li TR, Chen HM, Liu D (2013) Incremental approaches for updating approximations in set-valued ordered information systems. *Knowl-Based Syst* 50:218–233
- [27] Luo C, Li TR, Chen HM (2014) Dynamic maintenance of approximations in set-valued ordered decision systems under the attribute generalization. *Inf Sci* 257:210–228
- [28] Luo, C., Li, T., Chen, H., Lu, L.: Fast algorithms for computing rough approximations in set-valued decision systems while updating criteria values. *Inf. Sci.* 299, 221–242 (2015)
- [29] Wang, S., Li, T., Luo, C., Fujita, H.: Efficient updating rough approximations with multi-dimensional variation of ordered data. *Inf. Sci.* 372, 690–708 (2016)
- [30] Chen, H., Li, T., Ruan, D.: Maintenance of approximations in incomplete ordered decision systems while attribute values coarsening or refining. *Knowl. Based Syst.* 31, 140–161 (2012)
- [31] Luo, C., Li, T., Chen, H., Fujita, H., Yi, Z.: Incremental rough set approach for hierarchical multicriteria classification. *Inf. Sci.* 429, 72–87 (2018)
- [32] Sarra Bouzayane, Ines Saad, A Multicriteria Approach based on Rough Set Theory for the Incremental Periodic Prediction, *European Journal of Operational Research* (2020), doi: <https://doi.org/10.1016/j.ejor.2020.03.024>
- [33] Li, Y., Jin, Y., & Sun, X. (2015). Incremental method of updating approximations in DRSA under variations of multiple objects. *International Journal of Machine Learning and Cybernetics*, 9(2), 295–308. doi:10.1007/s13042-015-0477-8
- [34] Li, S., & Hong, Z. (2019). A Dynamic Dominance-Based Rough Set Approach for Processing Ordered Data. *Lecture Notes in Computer Science*, 312–320. doi:10.1007/978-3-030-22815-6_24
- [35] UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>].

Faryal Nosheen is a PhD student at National University of Sciences and Technology (NUST), Islamabad, Pakistan. She received M.S. degree in software engineering from NUST, Islamabad, Pakistan. Her research interests include machine learning, data sciences and rough set theory.

Usman Qamar is an associate professor and HOD of Department of Computer and Software Engineering, College of Electrical and Mechanical Engineering, NUST, Islamabad, Pakistan. He received his PhD degree in data sciences from University of Manchester. His research interests include data sciences, health informatics, artificial intelligence, and natural language processing.

Muhammad Summair Raza is an assistant professor at Department of Computer Science and Technology, Virtual University of Pakistan, Lahore, Pakistan. He received his PhD degree in software engineering from NUST, Islamabad, Pakistan. His research interests include big data, rough set theory and health informatics.