Fuzzy Sets and Systems 520 (2025) 109548

Contents lists available at ScienceDirect

Fuzzy Sets and Systems

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

Efficiency of decision rule sets in fuzzy rough set theory ☆

Fernando Chacón-Gómez *, M. Eugenia Cornejo, Jesús Medina

Department of Mathematics, University of Cádiz, Spain

A R T I C L E   I N F O

Keywords:
Fuzzy rough set theory
Decision rules
Decision algorithms
Efficiency

A B S T R A C T

Datasets have been interpreted in (fuzzy) rough set theory as decision tables to obtain useful information to be used, for example, in decision making. These tables have been modeled through a collection of decision rules, which was called decision algorithm by Pawlak. These algorithms are analyzed by the notion of efficiency, which evaluates their quality of classification. This paper presents two different approaches for defining the notion of efficiency in the fuzzy framework. The first approach is a direct generalization to the classical case, while the second one is focused on obtaining a bounded efficiency preserving the philosophy of the classical framework. Both approaches are illustrated by means of different properties and examples.

1. Introduction

Rough set theory (RST) [18,19] is a mathematical tool to obtain information from imprecise, incomplete and/or inconsistent databases. This last feature is one of the most difficult problems in classification tasks [5,15,17,22]. RST considers different research lines for handling it, such as, decision rules [4,24] for modeling the datasets, feature selection [12,13,25] for detecting the most relevant attributes, reducts [7,14] for also removing unnecessary information, and bireducts [1–3,23] for computing consistent subsets of the given databases, for instance.

Pawlak used decision rules in [18] to represent database information through the notion of decision algorithm. A decision algorithm is a set of non-zero supported and independent decision rules representing the objects and the consistency of the considered database. However, this notion has not received all the attention it deserves. Maybe, this is because of, in the classical case, an algorithm is straightforwardly obtained from a database considering a decision rule for each singular object in the database. This decision algorithm has mainly been, by default, the representative rule set of the database taken into account in the literature. Pawlak also introduced in [18] the definition of a parameter associated with the decision algorithm in order to represent its classifying role, which was called *efficiency of the decision algorithm*. This notion has not been taken much into account in subsequent works, mainly because the notion of decision algorithm was not used either, and they are completely related.

Nevertheless, these notions are very necessary when imperfect data is present in the dataset. For handling this kind of data, a fuzzy approach of rough set theory was presented in [10,16], and the fuzzy notions of support, independence, representativeness of the objects, maintenance of consistency and decision algorithm were introduced and studied in [6,9]. Now, the notion of efficiency of a decision algorithm must be extended to the fuzzy setting. In this paper, the natural extension of the original definition will be introduced. Moreover, a new normalized variant will be presented, that is, this new definition of efficiency will be bounded by 0 and 1.

☆ Partially supported by the project PID2019-108991GB-I00 funded by MICIU/AEI/10.13039/501100011033, the project PID2022-137620NB-I00 funded by MICIU/AEI/10.13039/501100011033 and FEDER, UE, by the grant TED2021-129748B-I00 funded by MCIN/AEI/10.13039/501100011033 and European Union NextGeneration EU/PRTR, and by the Research and Transfer Promotion Program of the University of Cádiz.

* Corresponding author.
E-mail addresses: fernando.chacon@uca.es (F. Chacón-Gómez), mariaeugenia.cornejo@uca.es (M.E. Cornejo), jesus.medina@uca.es (J. Medina).

https://doi.org/10.1016/j.fss.2025.109548
Received 22 April 2024; Received in revised form 3 June 2025; Accepted 29 July 2025

Available online 6 August 2025

For this purpose, a new definition of strength for decision rules has been introduced in the fuzzy rough set theory framework, which gives more relevance to the most representative rules of the algorithm. The consideration of a normalized efficiency is fundamental in order to compare decision algorithms from different datasets and detect the classificatory nature of each one.

The structure of the paper is supported by the following sections. Section 2 recalls several preliminary notions concerning to decision rules and decision algorithms in fuzzy rough set theory. Section 3 introduces two different approaches to study the efficiency of decision algorithms in the fuzzy framework. Finally, Section 4 presents some conclusions and prospects for future work.

2. Decision algorithms in fuzzy rough set theory

A classical complete decision algorithm is a set of independent and admissible decision rules whose antecedents and consequents cover the set of objects and that preserve the consistency of the decision table [20,21]. This notion was extended to the fuzzy framework in [9], where a detailed study about decision rules and their relevance indicators was carried out. This section is devoted to recalling the most important notions concerning complete decision algorithms in fuzzy rough set theory.

Definition 1 *([9]).* Let U and $\mathcal{A}$ be non-empty sets of objects and attributes, respectively. A *decision table* is a tuple $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ such that $\mathcal{A}_d = \mathcal{A} \cup \{d\}$ with $d \notin \mathcal{A}$, $\mathcal{V}_{\mathcal{A}_d} = \{V_a \mid a \in \mathcal{A}_d\}$, where V_a is the set of values associated with the attribute a over U, and $\overline{\mathcal{A}_d} = \{\bar{a} \mid a \in \mathcal{A}_d, \bar{a} : U \to V_a\}$. In this case, the attributes of $\mathcal{A}$ are called *condition attributes* and d is called *decision attribute*.

Decision tables allow us to represent databases in fuzzy rough set theory, and from them, it is possible to define different relations useful to compare either a pair of objects according to an attribute, or pairs of objects by using a subset of attributes instead of a unique attribute.

Definition 2 *([10]).* Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table and $B \subseteq \mathcal{A}$.

- An *a-indiscernibility relation* is a mapping $R_a : U \times U \to [0,1]$, with $a \in \mathcal{A}_d$, which is a fuzzy tolerance relation, that is, a reflexive and symmetrical fuzzy relation.
- A *B-indiscernibility relation* is a mapping $R_B : U \times U \to [0,1]$ defined, for each pair of objects $x, y \in U$, as:

  $$R_B(x,y) = @(\mathcal{R}_B^{x,y}(a_1), \dots, \mathcal{R}_B^{x,y}(a_m))$$

  where $\mathcal{A} = \{a_1, \dots, a_m\}$ and $\mathcal{R}_B^{x,y} : \mathcal{A} \to [0,1]$ is defined as follows, for each $a \in \mathcal{A}$:

  $$\mathcal{R}_B^{x,y}(a) = \begin{cases} R_a(x,y) & \text{if } a \in B \\ 1 & \text{otherwise} \end{cases}$$

  and $@ : [0,1]^m \to [0,1]$ is an aggregation operator, that is, an increasing operator on each argument satisfying $@(1,\dots,1) = 1$ and $@(0,\dots,0) = 0$.
- A *separable fuzzy tolerance relation* is a mapping $T_a : V_a \times V_a \to [0,1]$, with $a \in \mathcal{A}$, which is a fuzzy tolerance relation satisfying the next property:

  $$\text{If } T_a(v,w) = 1, \text{ then } v = w$$

Taking into account that all tolerance relations are reflexive, it is easy to conclude that all separable tolerance relations T_a satisfy that $T_a(v,w) = 1$ if and only if $v = w$.

Decision rules offer a logic representation of decision tables which makes easier their interpretation. A formal language is indispensable to describe the information contained in decision tables in logical terms, allowing us to compute how much an object satisfies a formula. These notions are formally introduced below.

Definition 3 *([9]).* Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table, $B \subseteq \mathcal{A}_d$, $C \subseteq \mathcal{A}$ and $T = \{T_a : V_a \times V_a \to [0,1] \mid a \in \mathcal{A}_d\}$ be a family of separable fuzzy tolerance relations.

- The *set of formulas associated with* B, denoted as $For(B)$, is built from attribute-value pairs (a, v), where $a \in B$ and $v \in V_a$, by means of the conjunction and disjunction logical connectives, $\wedge$ and $\vee$, respectively.
- The mapping $\|\cdot\|_S^T : For(B) \to [0,1]^U$ defined, for each $x \in U$ and $\Phi = (a, v)$, with $a \in B$ and $v \in V_a$, as:

  $$\|\Phi\|_S^T(x) = T_a(\bar{a}(x), v)$$

  is the *degree of satisfiability to the formula* Φ *of the object* x, through the relationships between the values of the attributes in the object x and the values of the attributes in the formula Φ.
- For every $\Phi, \Psi \in For(B)$, the *degree of satisfiability to the conjunction and disjunction of formulas* are defined, for each $x \in U$, as follows:

$$\|\Phi \wedge \Psi\|^T_S(x) = \min\{\|\Phi\|^T_S(x), \|\Psi\|^T_S(x)\}$$
$$\|\Phi \vee \Psi\|^T_S(x) = \max\{\|\Phi\|^T_S(x), \|\Psi\|^T_S(x)\}$$

- A *decision rule* in S is an expression $\Phi \to \Psi$, where $\Phi \in For(C)$ is the antecedent of the decision rule and $\Psi \in For(\{d\})$ is the consequent of the decision rule.
- *An object* $x \in U$ *fully supports the decision rule* $\Phi \to \Psi$ if$\|\Phi \wedge \Psi\|^T_S(x) = 1$. In this way, we can say that *the object* x *induces the decision rule* $\Phi \to \Psi$.

Different fuzzy relevance indicators, useful to give a description of decision rules, were presented in [9]. Specifically, this paper focuses on the T-support and the T-strength in order to draw the representativeness of decision rules in the decision table. Before presenting these fuzzy relevance indicators, we need to recall the notion of cardinal of a fuzzy set [11].

Definition 4 *([11])*. Given a universe V, the *cardinal of a fuzzy subset* $g : V \to [0,1]$ is defined as:

$$\text{card}_F(g) = \sum_{x \in V} g(x)$$

Definition 5 *([9])*. Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table, $\Phi \to \Psi$ be a decision rule in S and $T = \{T_a : V_a \times V_a \to [0,1] \mid a \in \mathcal{A}_d\}$ be a family of separable fuzzy tolerance relations. We call:

- T-*support* of the decision rule $\Phi \to \Psi$ to the value:
$$supp^T_S(\Phi, \Psi) = \text{card}_F(\|\Phi \wedge \Psi\|^T_S)$$
- T-*strength* of the decision rule $\Phi \to \Psi$ to the value:
$$\sigma^T_S(\Phi, \Psi) = \frac{supp^T_S(\Phi, \Psi)}{\text{card}(U)}$$
where card$(\cdot)$ denotes the cardinal of a classical set.

It is important to take into account that the T-support and the T-strength generalize the support and strength of the classical framework [9]. Next, we include the definition of a separable tolerance relation capable of comparing antecedents and consequents of decision rules. The former will be the conjunction of attribute-value pairs, whereas the latter will be an attribute-value pair, because only one decision attribute is considered in Definition 1.

Definition 6 *([9])*. Given a decision table $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$, a family of separable fuzzy tolerance relations $T = \{T_a : V_a \times V_a \to [0,1] \mid a \in \mathcal{A}_d\}$ and $\Phi, \Phi' \in For(\mathcal{A}_d)$, such that

$$\Phi = (a_1, v_1) \wedge \ldots \wedge (a_n, v_n) \quad \text{and} \quad \Phi' = (a'_1, w_1) \wedge \ldots \wedge (a'_m, w_m)$$

we define the F-*indiscernibility relation* as a separable fuzzy tolerance relation $R_{Fd} : For(\mathcal{A}_d) \times For(\mathcal{A}_d) \to [0,1]$ given by

$$R_{Fd}(\Phi, \Phi') = \begin{cases} \bigwedge_{i \in \{1,\ldots,n\}} T_{a_i}(v_i, w_i) & \text{if } n = m \text{ and } a_i = a'_i \text{ for all } i \in \{1,\ldots,n\} \\ 0 & \text{otherwise} \end{cases}$$

Given $\alpha \in [0,1]$, we define the R_{Fd}-α-*block*, for each $\Phi \in For(\mathcal{A}_d)$, as follows:

$$[\Phi]_\alpha = \{\Phi' \in For(\mathcal{A}_d) \mid \alpha \le R_{Fd}(\Phi, \Phi')\}$$

If $\Phi' \in [\Phi]_\alpha$, then we will say that Φ *and* Φ' *are* R_{Fd}-α-*related*.

The use of R_{Fd}-α-blocks is fundamental to summarize those formulas that are considered similar. Clearly, the R_{Fd}-α-blocks are decreasing in α, and $[\Phi]_\alpha = \{\Phi\}$ if T is a Boolean family and $\alpha > 0$. R_{Fd}-α-blocks also allow us to obtain a set of rules without contradictions among them, that is, a consistent set of rules.

Definition 7 *([9])*. Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table, $\alpha \in (0,1]$, $R_{Fd} : For(\mathcal{A}_d) \times For(\mathcal{A}_d) \to [0,1]$ be a F-indiscernibility relation and $Dec(S) = \{\Phi_i \to \Psi_i \mid i \in \{1,\ldots,m\}\}$, with $m \ge 2$,[1] be a set of decision rules in S. The set of α-*consistent decision rules* is defined as follows:

[1] In this paper, a minimum of two decision rules is considered in decision algorithms, as Pawlak stated [21], but notice that similar, albeit pointless, studies can be carried out in the trivial classes of one algorithm with only one decision rule.

$$Dec^+_\alpha(S) = \{\Phi \to \Psi \in Dec(S) \mid \text{if for each } \Phi' \to \Psi' \in Dec(S) \text{ such that } \Phi' \in [\Phi]_\alpha \text{ then } \Psi' \in [\Psi]_\alpha\}$$

The set of α-consistent decision rules is important to extract the most reliable information contained in decision tables, since it contains all the non-contradictory decision rules given the threshold α.

Another helpful notion to measure the consistency is the fuzzy positive region, which provides a degree of consistency of a given object in the decision table, by using indiscernibility relations and a fuzzy implication. This notion plays an important role in decision algorithms and it was framed in the multi-adjoint approach in [10]. For that reason, we need to fix the multi-adjoint property-oriented frame [10,16] employed in this work.

Definition 8 *([16]).* A *multi-adjoint property-oriented frame* is a tuple$(P_1, L_2, L_3, \&_1, \dots, \&_n)$ composed of a partially ordered set $(P_1, \preceq_1)$, two complete lattices $(L_2, \preceq_2)$ and $(L_3, \preceq_3)$, and adjoint triples $(\&_i, \swarrow^i, \nwarrow_i)$ with respect to P_1, L_2, L_3, for all $i \in \{1, \dots, n\}$, that is, the tuple $(\&_i, \swarrow^i, \nwarrow_i)$ satisfies the following double equivalence, for each $x \in P_1$, $y \in L_2$, $z \in L_3$:

$$x \preceq_1 z \swarrow^i y \quad \text{iff} \quad x \&_i y \preceq_3 z \quad \text{iff} \quad y \preceq_2 z \nwarrow_i x$$

The mapping $\&_i$ is called *adjoint conjunctor*, $\swarrow^i$ is called *left residuated fuzzy implication* of $\&$ and $\nwarrow_i$ is called *right residuated fuzzy implication* of $\&$.

Now, focusing on the multi-adjoint property-oriented frame $([0,1], [0,1], [0,1], \&_1, \dots, \&_n)$, we present the notion of fuzzy positive region when R_d is a Boolean d-indiscernibility relation given in [10].

Definition 9 *([10]).* Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table, R_d be a Boolean d-indiscernibility relation and the mapping $\tau \colon U \times U \to \{1, \dots, n\}$. The *multi-adjoint fuzzy $\mathcal{A}$-positive region* is defined, for each $y \in U$, as:

$$\mathrm{POS}^{\mathrm{f}}_{\mathcal{A}}(y) = \inf\{R_d(y,x) \nwarrow_{\tau(x,y)} R_{\mathcal{A}}(x,y) \mid x \in U\}$$

where $\nwarrow_{\tau(x,y)}$ is the right residuated fuzzy implication of $\&_{\tau(x,y)}$ associated with the pair of objects x, y.

Once the previous notions have been presented, we are in a position to introduce the definition of decision algorithm in the fuzzy rough set theory framework, which is a set of decision rules satisfying some requirements based on the previous fuzzy relevance indicators.

Definition 10 *([9]).* Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table, $\alpha \in (0,1]$, $\alpha_1, \alpha_2, \alpha_4 \in [0,1)$, $\alpha_3 \in (0, \mathrm{card}(U)]$, $T = \{T_a \colon V_a \times V_a \to [0,1] \mid a \in \mathcal{A}_d\}$ be a family of separable fuzzy tolerance relations and $Dec(S) = \{\Phi_i \to \Psi_i \mid i \in \{1, \dots, m\}\}$, with $m \geq 2$, be a set of decision rules in S. We say that:

1. $Dec(S)$ is a set of $\alpha_1\alpha_2$-*pairwise mutually exclusive (independent)* decision rules, if each pair of decision rules $\Phi \to \Psi, \Phi' \to \Psi' \in Dec(S)$ satisfies that $\Phi = \Phi'$ or $\|\Phi \wedge \Phi'\|^T_S(x) \leq \alpha_1$ and $\Psi = \Psi'$ or $\|\Psi \wedge \Psi'\|^T_S(x) \leq \alpha_2$, for all $x \in U$.
2. $Dec(S)$ *covers* U, if $\mathrm{card}_F(\|\bigvee_{i=1}^m \Phi_i\|^T_S) = \mathrm{card}_F(\|\bigvee_{i=1}^m \Psi_i\|^T_S) = \mathrm{card}(U)$.
3. The decision rule $\Phi \to \Psi \in Dec(S)$ is α_3-*admissible in* S if $\alpha_3 \leq supp^T_S(\Phi, \Psi)$.
4. $Dec(S)$ *preserves the α-consistency of S with a degree* α_4 if the next inequality holds for each $x \in U$:

$$\left| \mathrm{POS}^{\mathrm{f}}_{\mathcal{A}}(x) - \bigvee_{\Phi \to \Psi \in Dec^+_\alpha(S)} \|\Phi\|^T_S(x) \right| \leq \alpha_4$$

The set of decision rules $Dec(S)$ satisfying the previous properties for the values $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ is called $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-*decision algorithm in* S and it is denoted as $DA_T(S)$.

Furthermore, if for each $x \in U$ there exists $\Phi \to \Psi \in DA_T(S)$ such that $\|\Phi \wedge \Psi\|^T_S(x) = 1$, then $DA_T(S)$ is called a *complete* $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-*decision algorithm.*

Notice that, α_1, α_2 and α_4 cannot be 1, since the properties in which they are involved are always satisfied, and so they are valueless. The interpretation of the role of the previous thresholds can be consulted in [9].

It is also important to notice that the order of the rules in the set $DA_T(S)$ does not affect to any threshold. As a result, if a set of rules is a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm, then any reordering is also a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm. Moreover, it is possible to take appropriate values of each threshold α_i, with $i \in \{1,2,3,4\}$, so that a set of decision rules $Dec(S)$ where each rule is induced by an object of the decision table S is a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm, as it was shown in [9]. Finally, it is convenient to remark that complete $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithms of fuzzy rough set theory generalize complete decision algorithms of RST [9].

Decision rules belonging to classical complete decision algorithms satisfy interesting properties. In particular, the sum of the support of all the decision rules of a classical complete decision algorithm coincides with the number of objects considered in the decision table. This fact leads to that the sum of all the strengths is 1. This result is generalized to the fuzzy rough set theory framework next.

Lemma 11. *Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table, $R_\mathcal{A}$ and R_a be Boolean separable tolerance relations for all $a \in \mathcal{A}_d$ and $T = \{T_a : V_a \times V_a \to \{0,1\} \mid a \in \mathcal{A}_d\}$ be a family of Boolean separable tolerance relations. Given $DA_T(S)$ a complete $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm we obtain that*

$$\sum_{\Phi \to \Psi \in DA_T(S)} supp_S^T(\Phi, \Psi) = card(U)$$
$$\sum_{\Phi \to \Psi \in DA_T(S)} \sigma_S^T(\Phi, \Psi) = 1$$

Proof. The proof holds because complete $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithms, T-support and T-strength extend their corresponding classical definitions. □

3. Efficiency of $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithms

This section studies the quality of the classification of the objects belonging to a decision table provided by a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm, by means of two alternative definitions of efficiency. The notion of efficiency was originally introduced in the RST framework [21] to study classical complete decision algorithms, which is recalled below.

Definition 12 *([21]).* Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table and $DA(S)$ be a complete classical decision algorithm. We call *efficiency* of $DA(S)$ to the value:

$$\eta(DA(S)) = \sum_{\{\Phi \in For(\mathcal{A}) \mid \Phi \to \Psi \in DA(S)\}} \eta_\Phi(DA(S))$$

where $\eta_\Phi(DA(S)) = \max\{\sigma_S(\Phi, \Psi') \mid \Phi \to \Psi' \in DA(S)\}$ and $\sigma_S(\Phi, \Psi')$ is the classical strength.

According to the previous definition, the efficiency of a given classical $DA(S)$ is computed considering, for each antecedent, the greatest values of the strength for the rules in $DA(S)$ with such antecedent. Hence, the efficiency provides the proportion of objects classified by means of the most representative rule for each antecedent. Consequently, by the definition of the strength, the classical efficiency is bounded by 1. Furthermore, if all the antecedents of the rules in $DA(S)$ are different, then the efficiency is equal to the sum of the strengths of the rules, which is clearly 1 in the classical case.

3.1. *ε-efficiency of $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithms*

This section is devoted to presenting the natural extension of the efficiency to the fuzzy setting taking into account that the considered antecedents can be similar to others in $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithms.

Definition 13. Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table and $DA_T(S)$ be a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm. Given $\varepsilon \in [0,1]$, we call *ε-efficiency* of $DA_T(S)$ to the number:

$$\eta^\varepsilon(DA_T(S)) = \sum_{\{\Phi \in For(\mathcal{A}) \mid \Phi \to \Psi \in DA_T(S)\}} \eta_\Phi^\varepsilon(DA_T(S))$$

where $\eta_\Phi^\varepsilon(DA_T(S)) = \max\{\sigma_S^T(\Phi', \Psi') \mid \Phi' \to \Psi' \in DA_T(S), \Phi' \in [\Phi]_\varepsilon\}$.

Notice that, we have considered a threshold ε in order to compare the antecedents of the rules present in the algorithm by means of R_{Fd}-ε-blocks. In this way, if a higher threshold is chosen by the user, a more restrictive comparison between the antecedents is carried out, so that the values present in the antecedents have to be more similar in order to consider both antecedents as R_{Fd}-ε-related. Therefore, the threshold will supply a great level of flexibility to the study depending on the preferences of the user, and it will allow us to compare some ε-efficiencies of the same decision algorithm by considering different thresholds. Next result presents some properties obtained from Definition 13.

Proposition 14. *Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table and $DA_T(S) = \{\Phi_i \to \Psi_i \mid i \in \{1, \dots, m\}\}$, with $m \geq 2$, be a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm. The following properties hold:*

1. *The notion of ε-efficiency generalizes the classical one, that is if $T = \{T_a : V_a \times V_a \to \{0,1\} \mid a \in \mathcal{A}_d\}$ is a family of Boolean separable tolerance relations and $\varepsilon > 0$, then: $\eta^\varepsilon(DA_T(S)) = \eta(DA_T(S))$.*
2. *The ε-efficiency of $DA_T(S)$ is decreasing in ε, that is, given $\varepsilon_1, \varepsilon_2 \in [0,1]$, if $\varepsilon_1 \leq \varepsilon_2$ then*

$$\eta^{\varepsilon_2}(DA_T(S)) \leq \eta^{\varepsilon_1}(DA_T(S))$$

3. *If* $\max\{R_{Fd}(\Phi_i,\Phi_j) \mid i,j \in \{1,\dots,m\}, i \neq j\} < \varepsilon$*, then the* ε*-efficiency of* $DA_T(S)$ *is given as:*

$$\eta^{\varepsilon}(DA_T(S)) = \sum_{\Phi\to\Psi\in DA_T(S)} \sigma_S^T(\Phi,\Psi)$$

4. *Let* $\Phi_j \to \Psi_j \in DA_T(S)$ *such that*

$$\sigma_S^T(\Phi_j,\Psi_j) = \max\{\sigma_S^T(\Phi,\Psi) \mid \Phi\to\Psi \in DA_T(S)\}$$

If $\varepsilon \leq \min\{R_{Fd}(\Phi_i,\Phi_j) \mid i \in \{1,\dots,m\}\}$*, then the* ε*-efficiency of* $DA_T(S)$ *is given as:*

$$\eta^{\varepsilon}(DA_T(S)) = \sigma_S^T(\Phi_j,\Psi_j)\cdot card(\{\Phi \in For(\mathcal{A}) \mid \Phi\to\Psi \in DA_T(S)\})$$

Proof. First of all, we prove Item 1. Since the mappings T_a are Booleans and separable, for all $a \in \mathcal{A}_d$, we obtain that $\sigma_S^T(\Phi,\Psi) = \sigma_S(\Phi,\Psi)$, for all $\Phi\to\Psi \in DA_T(S)$. Moreover, since $\varepsilon > 0$, we have that $[\Phi]_\varepsilon = \{\Phi\}$, for all $\Phi \in For(\mathcal{A})$. As a consequence:

$$\begin{aligned}\eta^{\varepsilon}_{\Phi}(DA_T(S)) &= \max\{\sigma_S^T(\Phi',\Psi') \mid \Phi'\to\Psi' \in DA_T(S), \Phi' \in [\Phi]_\varepsilon\} = \max\{\sigma_S^T(\Phi,\Psi') \mid \Phi\to\Psi' \in DA_T(S)\}\\ &= \max\{\sigma_S(\Phi,\Psi') \mid \Phi\to\Psi' \in DA_T(S)\}\\ &= \eta_\Phi(DA_T(S))\end{aligned}$$

for all $\Phi \in For(\mathcal{A})$, with $\Phi\to\Psi \in DA_T(S)$. In conclusion,

$$\eta^{\varepsilon}(DA_T(S)) = \eta(DA_T(S))$$

Now, we focus on Item 2. Let $\varepsilon_1,\varepsilon_2 \in [0,1]$ such that $\varepsilon_1 \leq \varepsilon_2$ and $\Phi\to\Psi \in DA_T(S)$. Hence, $[\Phi]_{\varepsilon_2} \subseteq [\Phi]_{\varepsilon_1}$. As a result:

$$\{\Phi'\to\Psi' \in DA_T(S) \mid \Phi' \in [\Phi]_{\varepsilon_2}\} \subseteq \{\Phi''\to\Psi'' \in DA_T(S) \mid \Phi'' \in [\Phi]_{\varepsilon_1}\}$$

Therefore, by Definition 13, we deduce that:

$$\begin{aligned}\eta^{\varepsilon_2}_{\Phi}(DA_T(S)) &= \max\{\sigma_S^T(\Phi',\Psi') \mid \Phi'\to\Psi' \in DA_T(S), \Phi' \in [\Phi]_{\varepsilon_2}\}\\ &\leq \max\{\sigma_S^T(\Phi'',\Psi'') \mid \Phi''\to\Psi'' \in DA_T(S), \Phi'' \in [\Phi]_{\varepsilon_1}\}\\ &= \eta^{\varepsilon_1}_{\Phi}(DA_T(S))\end{aligned}$$

Applying again Definition 13, we obtain that:

$$\eta^{\varepsilon_2}(DA_T(S)) = \sum_{\{\Phi'''\in For(\mathcal{A}) \mid \Phi'''\to\Psi'''\in DA_T(S)\}} \eta^{\varepsilon_2}_{\Phi'''}(DA_T(S)) \leq \sum_{\{\Phi'''\in For(\mathcal{A}) \mid \Phi'''\to\Psi'''\in DA_T(S)\}} \eta^{\varepsilon_1}_{\Phi'''}(DA_T(S)) = \eta^{\varepsilon_1}(DA_T(S))$$

In conclusion, the ε-efficiency of a $(\alpha_1,\alpha_2,\alpha_3,\alpha_4)_\alpha$-decision algorithm is decreasing in the threshold ε.

For Item 3, since $\max\{R_{Fd}(\Phi_i,\Phi_j) \mid i,j \in \{1,\dots,m\}, i\neq j\} < \varepsilon$, by Definition 6, we obtain that $[\Phi]_\varepsilon = \{\Phi\}$, for each Φ antecedent of decision rules in $DA_T(S)$. Therefore, all the antecedents are different each other. As a consequence, by Definition 13, we have that:

$$\eta^{\varepsilon}_{\Phi}(DA_T(S)) = \max\{\sigma_S^T(\Phi',\Psi') \mid \Phi'\to\Psi' \in DA_T(S), \Phi' \in [\Phi]_\varepsilon\} = \sigma_S^T(\Phi,\Psi)$$

for all $\Phi \in For(\mathcal{A})$ such that $\Phi\to\Psi \in DA_T(S)$. Therefore, by Definition 13, we conclude that:

$$\eta^{\varepsilon}(DA_T(S)) = \sum_{\{\Phi\in For(\mathcal{A}) \mid \Phi\to\Psi\in DA_T(S)\}} \eta^{\varepsilon}_{\Phi}(DA_T(S)) = \sum_{\{\Phi\in For(\mathcal{A}) \mid \Phi\to\Psi\in DA_T(S)\}} \sigma_S^T(\Phi,\Psi) = \sum_{\Phi\to\Psi\in DA_T(S)} \sigma_S^T(\Phi,\Psi)$$

Finally, we prove Item 4. Since $\varepsilon \leq \min\{R_{Fd}(\Phi_i,\Phi_j) \mid i \in \{1,\dots,m\}\}$, by Definition 6, we obtain that $\Phi_j \in \left[\Phi_i\right]_\varepsilon$ for all $i \in \{1,\dots,m\}$. Hence,

$$\Phi_j \to \Psi_j \in \{\Phi\to\Psi \in DA_T(S) \mid \Phi \in \left[\Phi_i\right]_\varepsilon\}$$

for all $i \in \{1,\dots,m\}$. Moreover, since $\Phi_j\to\Psi_j$ has the greatest value of T-strength, by Definition 13, the following chain of equalities is deduced, for each $\Phi \in For(\mathcal{A})$ such that $\Phi\to\Psi \in DA_T(S)$:

$$\eta^{\varepsilon}_{\Phi}(DA_T(S)) = \max\{\sigma_S^T(\Phi',\Psi') \mid \Phi'\to\Psi' \in DA_T(S), \Phi' \in [\Phi]_\varepsilon\} = \sigma_S^T(\Phi_j,\Psi_j)$$

As a consequence, by Definition 13, we conclude that:

$$\eta^{\varepsilon}(DA_T(S)) = \sum_{\{\Phi\in For(\mathcal{A}) \mid \Phi\to\Psi\in DA_T(S)\}} \eta^{\varepsilon}_{\Phi}(DA_T(S)) = \sigma_S^T(\Phi_j,\Psi_j)\cdot \text{card}(\{\Phi \in For(\mathcal{A}) \mid \Phi\to\Psi \in DA_T(S)\}) \quad \square$$

The interpretation of the previous properties and illustrative examples of Definition 13 can be consulted in [6,8]. The main inconvenient of Definition 13 is that the ε-efficiency of an algorithm is not bounded by 1, making difficult to compare algorithms from different datasets. This fact is due to two reasons: (1) the sum of the T-strength values of all decision rules of a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm can be greater than or equal to 1; (2) the decision rules can be considered more than once in the computation of the ε-efficiency, as Proposition 14(4) states. These problems are addressed in the next section with a new proposal.

3.2. Normalized ε-efficiency of $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithms

The given notions of T-strength and ε-efficiency are not completely adequate to analyze decision algorithms, then a normalized efficiency is required. In this section, we will introduce alternative definitions to these ones in order to solve the aforementioned problems. First of all, we present the notion of ε-T-strength of a decision rule considering the given set of decision rules, instead of the set of objects U, as done in Definition 5. Moreover, the consideration of the threshold ε allows us to set different criteria in the comparison between the studied decision rule and all the ones present in the given set.

Definition 15. Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table and $Dec(S)$ be a set of decision rules. Given $\varepsilon \in [0,1]$, we call *ε-T-strength* of the decision rule $\Phi \to \Psi$ with respect to $Dec(S)$ to the value:

$$\sigma_S^{\varepsilon,T}(\Phi, \Psi) = \frac{\displaystyle\sum_{\{\Phi' \to \Psi' \in Dec(S) \mid \Phi' \in [\Phi]_\varepsilon, \Psi' \in [\Psi]_\varepsilon\}} supp_S^T(\Phi', \Psi')}{\displaystyle\sum_{\Phi'' \to \Psi'' \in Dec(S)} supp_S^T(\Phi'', \Psi'')}$$

The main goal of introducing Definition 15 is to give more representativeness in the algorithm to those decision rules whose antecedents and consequents are R_{Fd}-ε-related to the antecedents and consequents of other decision rules of the algorithm. In this way, given the threshold ε, if we cannot discern either the antecedent or the consequent of a rule from other antecedents and consequents, then that rule must gain importance in the algorithm. Next result presents some properties deduced from Definition 15.

Proposition 16. *Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table, $Dec(S)$ be a set of decision rules and $\varepsilon \in [0,1]$. The following properties hold:*

1. *If $T = \{T_a : V_a \times V_a \to [0,1] \mid a \in \mathcal{A}_d\}$ is a family of Boolean tolerance relations, $R_\mathcal{A}$ and R_a are Boolean separable tolerance relations for all $a \in \mathcal{A}_d$, $\varepsilon > 0$, and $Dec(S)$ is a complete $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm then*

$$\sigma_S^{\varepsilon,T}(\Phi, \Psi) = \sigma_S(\Phi, \Psi), \quad \text{for all} \quad \Phi \to \Psi \in Dec(S)$$

2. *The ε-T-strength of a decision rule $\Phi \to \Psi$ with respect to $Dec(S)$ is decreasing in ε, that is, given $\varepsilon_1, \varepsilon_2 \in [0,1]$,*

$$\text{If} \quad \varepsilon_1 \le \varepsilon_2 \quad \text{then} \quad \sigma_S^{\varepsilon_2,T}(\Phi, \Psi) \le \sigma_S^{\varepsilon_1,T}(\Phi, \Psi)$$

3. *Given $\Phi \to \Psi \in Dec(S)$, if*

$$\varepsilon \le \min\{\min\{R_{Fd}(\Phi, \Phi'), R_{Fd}(\Psi, \Psi')\} \mid \Phi' \to \Psi' \in Dec(S)\}$$

then

$$\sigma_S^{\varepsilon,T}(\Phi, \Psi) = 1$$

4. *Given $\Phi \to \Psi \in Dec(S)$, if*

$$\max\{\min\{R_{Fd}(\Phi, \Phi'), R_{Fd}(\Psi, \Psi')\} \mid \Phi' \to \Psi' \in Dec(S) \setminus \{\Phi \to \Psi\}\} < \varepsilon$$

then

$$\sigma_S^{\varepsilon,T}(\Phi, \Psi) = \frac{supp_S^T(\Phi, \Psi)}{\displaystyle\sum_{\Phi' \to \Psi' \in Dec(S)} supp_S^T(\Phi', \Psi')}$$

Proof. First of all, we prove Item 1. Given $\varepsilon > 0$, as T is a family of Boolean tolerance relations, we obtain that $[\Phi]_\varepsilon = \{\Phi\}$ and $[\Psi]_\varepsilon = \{\Psi\}$, for all $\Phi \in For(\mathcal{A})$ and $\Psi \in For(\{d\})$. Therefore, applying Lemma 11, the following equality holds:

$$\sum_{\Phi \to \Psi \in Dec(S)} supp_S^T(\Phi, \Psi) = \text{card}(U)$$

Since we are in the classical case, for each $\Phi \to \Psi \in Dec(S)$, $supp_S^T(\Phi,\Psi) = supp_S(\Phi,\Psi)$ and, as a consequence, we conclude that:

$$\begin{aligned}\sigma_S^{\varepsilon,T}(\Phi,\Psi) &= \frac{\sum\limits_{\{\Phi'\to\Psi'\in Dec(S)\mid \Phi'\in[\Phi]_\varepsilon,\Psi'\in[\Psi]_\varepsilon\}} supp_S^T(\Phi',\Psi')}{\sum\limits_{\Phi''\to\Psi''\in Dec(S)} supp_S^T(\Phi,\Psi)} = \frac{supp_S^T(\Phi,\Psi)}{\sum\limits_{\Phi''\to\Psi''\in Dec(S)} supp_S^T(\Phi,\Psi)}\\ &= \frac{supp_S(\Phi,\Psi)}{\sum\limits_{\Phi''\to\Psi''\in Dec(S)} supp_S(\Phi,\Psi)} = \frac{supp_S(\Phi,\Psi)}{\mathrm{card}(U)}\\ &= \sigma_S(\Phi,\Psi)\end{aligned}$$

Now, we focus on Item 2. Let $\varepsilon_1, \varepsilon_2 \in [0,1]$ such that $\varepsilon_1 \le \varepsilon_2$. Consequently, $[\Phi]_{\varepsilon_2} \subseteq [\Phi]_{\varepsilon_1}$ and $[\Psi]_{\varepsilon_2} \subseteq [\Psi]_{\varepsilon_1}$, for all Φ and Ψ antecedent and consequent, respectively, of decision rules of $Dec(S)$. Therefore, for each decision rule $\Phi \to \Psi \in Dec(S)$ we obtain that:

$$\{\Phi' \to \Psi' \in Dec(S) \mid \Phi' \in [\Phi]_{\varepsilon_2}, \Psi' \in [\Psi]_{\varepsilon_2}\} \subseteq \{\Phi'' \to \Psi'' \in Dec(S) \mid \Phi'' \in [\Phi]_{\varepsilon_1}, \Psi'' \in [\Psi]_{\varepsilon_1}\}$$

and, as a result, the following inequality is satisfied:

$$\sum_{\{\Phi'\to\Psi'\in Dec(S)\mid \Phi'\in[\Phi]_{\varepsilon_2},\Psi'\in[\Psi]_{\varepsilon_2}\}} supp_S^T(\Phi',\Psi') \le \sum_{\{\Phi''\to\Psi''\in Dec(S)\mid \Phi''\in[\Phi]_{\varepsilon_1},\Psi''\in[\Psi]_{\varepsilon_1}\}} supp_S^T(\Phi'',\Psi'')$$

for all $\Phi \to \Psi \in Dec(S)$. In conclusion, $\sigma_S^{\varepsilon_2,T}(\Phi,\Psi) \le \sigma_S^{\varepsilon_1,T}(\Phi,\Psi)$ for all $\Phi \to \Psi \in Dec(S)$, that is, the ε-T-strength of a decision rule $\Phi \to \Psi$ with respect to $Dec(S)$ is decreasing in the threshold ε.

Next, Item 3 is proven. Let $\Phi \to \Psi \in Dec(S)$ and

$$\varepsilon \le \min\{\min\{R_{Fd}(\Phi,\Phi'), R_{Fd}(\Psi,\Psi')\} \mid \Phi' \to \Psi' \in Dec(S)\}$$

By Definition 6, for each $\Phi' \to \Psi' \in Dec(S)$, we have that $\Phi' \in [\Phi]_\varepsilon$ and $\Psi' \in [\Psi]_\varepsilon$. Therefore,

$$\{\Phi' \to \Psi' \in Dec(S) \mid \Phi' \in [\Phi]_\varepsilon, \Psi' \in [\Psi]_\varepsilon\} = Dec(S)$$

Consequently, by Definition 15, we conclude that:

$$\sigma_S^{\varepsilon,T}(\Phi,\Psi) = \frac{\sum\limits_{\{\Phi'\to\Psi'\in Dec(S)\mid \Phi'\in[\Phi]_\varepsilon,\Psi'\in[\Psi]_\varepsilon\}} supp_S^T(\Phi',\Psi')}{\sum\limits_{\Phi''\to\Psi''\in Dec(S)} supp_S^T(\Phi'',\Psi'')} = \frac{\sum\limits_{\Phi''\to\Psi''\in Dec(S)} supp_S^T(\Phi'',\Psi'')}{\sum\limits_{\Phi''\to\Psi''\in Dec(S)} supp_S^T(\Phi'',\Psi'')} = 1$$

Finally, we prove Item 4. Let $\Phi \to \Psi \in Dec(S)$ and

$$\varepsilon > \max\{\min\{R_{Fd}(\Phi,\Phi'), R_{Fd}(\Psi,\Psi')\} \mid \Phi' \to \Psi' \in Dec(S) \setminus \{\Phi \to \Psi\}\}$$

By Definition 6, for each $\Phi' \to \Psi' \in Dec(S) \setminus \{\Phi \to \Psi\}$, we have that either $\Phi' \notin [\Phi]_\varepsilon$ or $\Psi' \notin [\Psi]_\varepsilon$. As a consequence,

$$\{\Phi'' \to \Psi'' \in Dec(S) \mid \Phi'' \in [\Phi]_\varepsilon, \Psi'' \in [\Psi]_\varepsilon\} = \{\Phi \to \Psi\}$$

Therefore, by Definition 15, we obtain that:

$$\sigma_S^{\varepsilon,T}(\Phi,\Psi) = \frac{\sum\limits_{\{\Phi''\to\Psi''\in Dec(S)\mid \Phi''\in[\Phi]_\varepsilon,\Psi''\in[\Psi]_\varepsilon\}} supp_S^T(\Phi'',\Psi'')}{\sum\limits_{\Phi'''\to\Psi'''\in Dec(S)} supp_S^T(\Phi''',\Psi''')} = \frac{supp_S^T(\Phi,\Psi)}{\sum\limits_{\Phi'''\to\Psi'''\in Dec(S)} supp_S^T(\Phi''',\Psi''')} \qquad \square$$

Now, it is convenient to make some remarks about these properties. From Proposition 16(1) it is deduced that Definition 15 generalizes the classical notion of strength. Proposition 16(2) shows the monotonicity in the threshold ε. In this way, if we increase the value of ε then the antecedents and consequents of the rules have to be more similar to ones of the rule studied in order to consider them as R_{Fd}-ε-related. As a consequence, a more restrictive comparison reduces the ε-T-strength of each decision rule.

The last two items simplify the computation of the ε-T-strength of decision rules with respect to $Dec(S)$ in some particular cases and provide its boundaries. In Proposition 16(3), due to the choice of the threshold ε, we are unable to discern the decision rule $\Phi \to \Psi$ from the rest of rules. As a consequence, this rule has a complete representativeness in the set of rules, that is, its ε-T-strength with respect to $Dec(S)$ is 1. Proposition 16(4) focuses on the opposite case, that is, we can discern the rule $\Phi \to \Psi$ from the rest of rules of $Dec(S)$. Hence, its representativeness in $Dec(S)$ is only given by itself.

Finally, a new proposal to define the efficiency of a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm will be presented. Due to the ε-T-strength of two different rules can consider the same rule twice, once in the computation of each strength, we cannot directly consider the ε-T-strength in the computation of the efficiency, that is, we should introduce a mechanism to avoid repetitions.

Given a decision table $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$, a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm $DA_T(S) = \{\Phi_i \to \Psi_i \mid i \in \{1, \dots, m\}\}$, with $m \geq 2$, and $\varepsilon \in [0,1]$, we define the following value based on the strength as usual:

$$\eta^{N_\varepsilon}_{\Phi_i}(DA_T(S)) = \max\{\sigma^{\varepsilon,T}_S(\Phi, \Psi) \mid \Phi \to \Psi \in DA_T(S), \Phi \in [\Phi_i]_\varepsilon\}$$

We have introduced the letter N in the superscript in relation to the word "normal". Now, we will consider the following sets to avoid repetitions in the computation of the efficiency. The first set B_{Φ_i,Ψ_i} selects the rules in the algorithm maximizing $\eta^{N_\varepsilon}_{\Phi_i}$, that is, the rules from which the value $\eta^{N_\varepsilon}_{\Phi_i}$ is obtained. Specifically, we have:

$$B_{\Phi_i,\Psi_i} = \{\Phi' \to \Psi' \in DA_T(S) \mid \eta^{N_\varepsilon}_{\Phi_i}(DA_T(S)) = \sigma^{\varepsilon,T}_S(\Phi', \Psi'), \Phi' \in [\Phi_i]_\varepsilon\}$$

The second set contains the set of rules used in the computation of the ε-T-strength of the rule in B_{Φ_i,Ψ_i} with less subscript, that is,

$$C_{\Phi_i,\Psi_i} = \{\Phi'' \to \Psi'' \in DA_T(S) \mid \Phi'' \in [\Phi_k]_\varepsilon, \Psi'' \in [\Psi_k]_\varepsilon, k = \min\{l \in \{1, \dots, m\} \mid \Phi_l \to \Psi_l \in B_{\Phi_i,\Psi_i}\}\}$$

The third set takes into account the rules used in the computation of the strength only once. Specifically, it starts at the rule with the least subscript, and goes through the other rules in increasing subscript order up to the greatest one, including the rules used in the computation of the strength of the considered rule that have not been considered previously, that is

$$D_{\Phi_i,\Psi_i} = C_{\Phi_i,\Psi_i} \setminus \{\Phi'' \to \Psi'' \in D_{\Phi_j,\Psi_j} \mid j < i\}$$

From these sets we introduce the new normalized notion of efficiency.

Definition 17. Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table and $DA_T(S) = \{\Phi_i \to \Psi_i \mid i \in \{1, \dots, m\}\}$, with $m \geq 2$, a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm. Given $\varepsilon \in [0,1]$, we call *Normalized ε-efficiency*, denoted as *N_ε-efficiency*, of the $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm $DA_T(S)$ to the value:

$$\eta^{N_\varepsilon}(DA_T(S)) = \frac{\sum_{i=1}^{m} \sum_{\Phi'' \to \Psi'' \in D_{\Phi_i,\Psi_i}} supp^T_S(\Phi'', \Psi'')}{\sum_{\Phi''' \to \Psi''' \in DA_T(S)} supp^T_S(\Phi''', \Psi''')}$$

Definition 17 is introduced to bound the efficiency of a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm by 1 and to consider each decision rule of the algorithm at most once in the computations, which will overcome the main disadvantage of Definition 13. Moreover, the N_ε-efficiency also extends the classical notion of efficiency and its lower bound to the fuzzy framework, facilitating the interpretation of the obtained values.

Proposition 18. *Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table and $DA_T(S) = \{\Phi_i \to \Psi_i \mid i \in \{1, \dots, m\}\}$, with $m \geq 2$, a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm.*

1. *If $T = \{T_a : V_a \times V_a \to [0,1] \mid a \in \mathcal{A}_d\}$ is a family of Boolean tolerance relations, $R_{\mathcal{A}}$ and R_a are Boolean separable tolerance relations for all $a \in \mathcal{A}_d$, $\varepsilon > 0$, and $DA_T(S)$ is a complete $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm then $\eta^{N_\varepsilon}(DA_T(S)) = \eta(DA_T(S))$.*
2. *The following inequality holds:*

$$\max\{\sigma^{\varepsilon,T}_S(\Phi, \Psi) \mid \Phi \to \Psi \in DA_T(S)\} \leq \eta^{N_\varepsilon}(DA_T(S)) \leq 1$$

for all $\varepsilon \in [0,1]$.

Proof. First of all, we prove Item 1. Since $\varepsilon > 0$ and T is a family of Boolean tolerance relations, $[\Phi]_\varepsilon = \{\Phi\}$ and $[\Psi]_\varepsilon = \{\Psi\}$ for all $\Phi \to \Psi \in DA_T(S)$. In addition, by Proposition 16(1), we obtain that $\sigma^{\varepsilon,T}_S(\Phi, \Psi) = \sigma_S(\Phi, \Psi)$, for all $\Phi \to \Psi \in DA_T(S)$. Hence,

$$\begin{aligned}
\eta^{N_\varepsilon}_{\Phi_i}(DA_T(S)) &= \max\{\sigma^{\varepsilon,T}_S(\Phi, \Psi) \mid \Phi \to \Psi \in DA_T(S), \Phi \in [\Phi_i]_\varepsilon\} = \max\{\sigma^{\varepsilon,T}_S(\Phi_i, \Psi) \mid \Phi_i \to \Psi \in DA_T(S)\} \\
&= \max\{\sigma_S(\Phi_i, \Psi) \mid \Phi_i \to \Psi \in DA_T(S)\} \\
&= \eta_{\Phi_i}(DA_T(S))
\end{aligned}$$

for all $i \in \{1, \dots, m\}$. On the other hand, since $DA_T(S)$ is a complete $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm, by Corollary 11 we obtain that:

$$\sum_{\Phi \to \Psi \in DA_T(S)} supp^T_S(\Phi, \Psi) = \text{card}(U)$$

Now, we have to discern two different cases:

- Suppose that $\Phi_i \neq \Phi_j$, for all $i, j \in \{1, \ldots, m\}$ with $i \neq j$. Hence, by Definition 17, $B_{\Phi_i,\Psi_i} = C_{\Phi_i,\Psi_i} = D_{\Phi_i,\Psi_i} = \{\Phi_i \to \Psi_i\}$, for all $i \in \{1, \ldots, m\}$. Furthermore, taking into account that $\sigma_S(\Phi_i, \Psi_i) = \eta_{\Phi_i}(DA_T(S))$, for all $i \in \{1, \ldots, m\}$, we conclude that:

$$\begin{aligned}\eta^{N_\varepsilon}(DA_T(S)) &= \frac{\sum_{i=1}^{m} \sum_{\Phi'' \to \Psi'' \in D_{\Phi_i,\Psi_i}} supp_S^T(\Phi'', \Psi'')}{\sum_{\Phi''' \to \Psi''' \in DA_T(S)} supp_S^T(\Phi''', \Psi''')} = \frac{\sum_{i=1}^{m} supp_S^T(\Phi_i, \Psi_i)}{\sum_{\Phi''' \to \Psi''' \in DA_T(S)} supp_S^T(\Phi''', \Psi''')} = \sum_{\Phi'' \to \Psi'' \in DA_T(S)} \frac{supp_S(\Phi'', \Psi'')}{\text{card}(U)}\\ &= \sum_{\Phi'' \to \Psi'' \in DA_T(S)} \sigma_S(\Phi'', \Psi'') = \sum_{\{\Phi \in For(\mathcal{A}) \mid \Phi \to \Psi \in DA_T(S)\}} \eta_\Phi(DA_T(S))\\ &= \eta(DA_T(S))\end{aligned}$$

- Now, we suppose that there exist $i, j \in \{1, \ldots, m\}$, with $i \neq j$, such that $\Phi_j = \Phi_i$. Without loss of generality, we suppose that $i < j$. By Definition 17, we deduce that

$$\begin{aligned}\eta_{\Phi_j}^{N_\varepsilon}(DA_T(S)) &= \max\{\sigma_S^{\varepsilon,T}(\Phi, \Psi) \mid \Phi \to \Psi \in DA_T(S), \Phi \in [\Phi_j]_\varepsilon\}\\ &= \max\{\sigma_S^{\varepsilon,T}(\Phi, \Psi) \mid \Phi \to \Psi \in DA_T(S), \Phi \in [\Phi_i]_\varepsilon\}\\ &= \eta_{\Phi_i}^{N_\varepsilon}(DA_T(S))\\ B_{\Phi_j,\Psi_j} &= \{\Phi' \to \Psi' \in DA_T(S) \mid \eta_{\Phi_j}^{N_\varepsilon}(DA_T(S)) = \sigma_S^{\varepsilon,T}(\Phi', \Psi'), \Phi' \in [\Phi_j]_\varepsilon\}\\ &= \{\Phi' \to \Psi' \in DA_T(S) \mid \eta_{\Phi_i}^{N_\varepsilon}(DA_T(S)) = \sigma_S^{\varepsilon,T}(\Phi', \Psi'), \Phi' \in [\Phi_i]_\varepsilon\}\\ &= B_{\Phi_i,\Psi_i}\\ C_{\Phi_j,\Psi_j} &= \{\Phi'' \to \Psi'' \in DA_T(S) \mid \Phi'' \in [\Phi_k]_\varepsilon, \Psi'' \in [\Psi_k]_\varepsilon, k = \min\{l \in \{1, \ldots, m\} \mid \Phi_l \to \Psi_l \in B_{\Phi_j,\Psi_j}\}\}\\ &= \{\Phi'' \to \Psi'' \in DA_T(S) \mid \Phi'' \in [\Phi_k]_\varepsilon, \Psi'' \in [\Psi_k]_\varepsilon, k = \min\{l \in \{1, \ldots, m\} \mid \Phi_l \to \Psi_l \in B_{\Phi_i,\Psi_i}\}\}\\ &= C_{\Phi_i,\Psi_i}\end{aligned}$$

  Therefore, by Definition 17, for each $\Phi'' \to \Psi'' \in C_{\Phi_j,\Psi_j}$ we have that $\Phi'' \to \Psi'' \in D_{\Phi_k,\Psi_k}$, with $k \in \{1, \ldots, i\}$. In conclusion, $D_{\Phi_j,\Psi_j} = \varnothing$.
  Reciprocally, we suppose that there exists $j \in \{1, \ldots, m\}$ such that $D_{\Phi_j,\Psi_j} = \varnothing$. Therefore, for each $\Phi'' \to \Psi'' \in C_{\Phi_j,\Psi_j}$ there exists $i \in \{1, \ldots, j-1\}$ such that $\Phi'' \to \Psi'' \in D_{\Phi_i,\Psi_i} \subseteq C_{\Phi_i,\Psi_i}$. On the other hand, given $k \in \{1, \ldots, m\}$, we obtain that, if $\Phi'' \to \Psi'' \in C_{\Phi_k,\Psi_k}$ then $\Phi'' \in [\Phi_k]_\varepsilon$, that is, $\Phi'' = \Phi_k$. As a consequence, we conclude that $\Phi_j = \Phi_i$. In short, we have proven that there exist $i, j \in \{1, \ldots, m\}$, with $i < j$, such that $\Phi_i = \Phi_j$ if and only if $D_{\Phi_j,\Psi_j} = \varnothing$.
  Furthermore, notice that, given $i \in \{1, \ldots, m\}$, if $D_{\Phi_i,\Psi_i} \neq \varnothing$, then $\text{card}(D_{\Phi_i,\Psi_i}) = 1$. Indeed, if a pair of decision rules $\Phi''_j \to \Psi''_j, \Phi''_k \to \Psi''_k \in D_{\Phi_i,\Psi_i}$, then $\Phi''_j \to \Psi''_j, \Phi''_k \to \Psi''_k \in C_{\Phi_i,\Psi_i}$. As a consequence, there exists $\Phi''_l \to \Psi''_l \in B_{\Phi_i,\Psi_i}$ such that $\Phi''_j, \Phi''_k \in [\Phi''_l]_\varepsilon$ and $\Psi''_j, \Psi''_k \in [\Psi''_l]_\varepsilon$, that is, $\Phi''_l \to \Psi''_l = \Phi''_k \to \Psi''_k = \Phi''_j \to \Psi''_j$. Hence, we deduce that, if $\Phi'' \to \Psi'' \in D_{\Phi_i,\Psi_i}$ then $\{\Phi'' \to \Psi''\} = D_{\Phi_i,\Psi_i} \subseteq C_{\Phi_i,\Psi_i} \subseteq B_{\Phi_i,\Psi_i}$ and:

$$\begin{aligned}\frac{\sum_{\Phi'' \to \Psi'' \in D_{\Phi_i,\Psi_i}} supp_S^T(\Phi'', \Psi'')}{\sum_{\Phi''' \to \Psi''' \in DA_T(S)} supp_S^T(\Phi''', \Psi''')} &= \frac{supp_S^T(\Phi'', \Psi'')}{\sum_{\Phi''' \to \Psi''' \in DA_T(S)} supp_S^T(\Phi''', \Psi''')}\\ &= \sigma_S^{\varepsilon,T}(\Phi'', \Psi'') = \eta_{\Phi_i}^{N_\varepsilon}(DA_T(S))\\ &= \eta_{\Phi_i}(DA_T(S))\end{aligned}$$

  Taking into account the previous reasonings, we obtain the final equality of Item 1.

Now, we will prove Item 2. Given $\varepsilon \in [0, 1]$, let $\Phi_j \to \Psi_j \in DA_T(S)$ such that $\sigma_S^{\varepsilon,T}(\Phi_j, \Psi_j) = \max\{\sigma_S^{\varepsilon,T}(\Phi, \Psi) \mid \Phi \to \Psi \in DA_T(S)\}$ with $j = \min\{j_1, \ldots, j_n\}$ such that $\sigma_S^{\varepsilon,T}(\Phi_j, \Psi_j) = \sigma_S^{\varepsilon,T}(\Phi_{j_i}, \Psi_{j_i})$ with $i \in \{1, \ldots, n\}$. For this reason, we obtain that:

$$\begin{aligned}C_{\Phi_j,\Psi_j} &= \{\Phi'' \to \Psi'' \in DA_T(S) \mid \Phi'' \in [\Phi_k]_\varepsilon, \Psi'' \in [\Psi_k]_\varepsilon, k = \min\{l \in \{1, \ldots, m\} \mid \Phi_l \to \Psi_l \in B_{\Phi_j,\Psi_j}\}\}\\ &= \{\Phi'' \to \Psi'' \in DA_T(S) \mid \Phi'' \in [\Phi_j]_\varepsilon, \Psi'' \in [\Psi_j]_\varepsilon\}\end{aligned}$$

On the other hand, notice that:

$$C_{\Phi_r,\Psi_r} \subseteq \bigcup_{i=1}^{m} D_{\Phi_i,\Psi_i}$$

for all $r \in \{1,\dots,m\}$. Indeed, let $\Phi \to \Psi \in C_{\Phi_r,\Psi_r}$. By Definition 17, either $\Phi \to \Psi \in D_{\Phi_r,\Psi_r}$ or there exists $i < r$ such that $\Phi \to \Psi \in D_{\Phi_i,\Psi_i}$. In any case, $\Phi \to \Psi \in \bigcup_{i=1}^{m} D_{\Phi_i,\Psi_i}$, so $C_{\Phi_r,\Psi_r} \subseteq \bigcup_{i=1}^{m} D_{\Phi_i,\Psi_i}$ for all $r \in \{1,\dots,m\}$. In conclusion,

$$\sigma_S^{\varepsilon,T}(\Phi_j,\Psi_j) = \frac{\sum\limits_{\{\Phi''\to\Psi''\in DA_T(S) \mid \Phi''\in[\Phi_j]_\varepsilon, \Psi''\in[\Psi_j]_\varepsilon\}} supp_S^T(\Phi'',\Psi'')}{\sum\limits_{\Phi'''\to\Psi'''\in DA_T(S)} supp_S^T(\Phi''',\Psi''')} = \frac{\sum\limits_{\Phi''\to\Psi''\in C_{\Phi_j,\Psi_j}} supp_S^T(\Phi'',\Psi'')}{\sum\limits_{\Phi'''\to\Psi'''\in DA_T(S)} supp_S^T(\Phi''',\Psi''')}$$

$$\leq \frac{\sum\limits_{i=1}^{m}\sum\limits_{\Phi''\to\Psi''\in D_{\Phi_i,\Psi_i}} supp_S^T(\Phi'',\Psi'')}{\sum\limits_{\Phi'''\to\Psi'''\in DA_T(S)} supp_S^T(\Phi''',\Psi''')}$$

$$= \eta^{N_\varepsilon}(DA_T(S))$$

Therefore, $\max\{\sigma_S^{\varepsilon,T}(\Phi,\Psi) \mid \Phi\to\Psi \in DA_T(S)\} \leq \eta^{N_\varepsilon}(DA_T(S))$. For the other inequality, notice that, given $i,j \in \{1,\dots,m\}$ such that $i<j$, if $\Phi\to\Psi \in D_{\Phi_i,\Psi_i}$ then $\Phi\to\Psi \notin D_{\Phi_j,\Psi_j}$, that is, no decision rule is considered twice. As a consequence,

$$\sum_{i=1}^{m}\sum_{\Phi\to\Psi\in D_{\Phi_i,\Psi_i}} supp_S^T(\Phi,\Psi) \leq \sum_{\Phi\to\Psi\in DA_T(S)} supp_S^T(\Phi,\Psi)$$

In conclusion, for all $\varepsilon \in [0,1]$, we have that:

$$\max\{\sigma_S^{\varepsilon,T}(\Phi,\Psi) \mid \Phi\to\Psi \in DA_T(S)\} \leq \eta^{N_\varepsilon}(DA_T(S)) \leq 1 \quad \square$$

Next result presents a particular case where the N_ε-efficiency of a decision algorithm is determined by the maximum of the ε-T-strengths of the decision rules belonging to the algorithm under consideration. By Proposition 18(2), we can guarantee that the mentioned case provides the minimum possible N_ε-efficiency for any algorithm.

Proposition 19. *Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table, $DA_T(S) = \{\Phi_i \to \Psi_i \mid i \in \{1,\dots,m\}\}$, with $m \geq 2$, a $(\alpha_1,\alpha_2,\alpha_3,\alpha_4)_\alpha$-decision algorithm, and $\Phi_j \to \Psi_j \in DA_T(S)$ such that $\sigma_S^{\varepsilon,T}(\Phi_j,\Psi_j) = \max\{\sigma_S^{\varepsilon,T}(\Phi,\Psi) \mid \Phi\to\Psi\in DA_T(S)\}$ and $j = \min\{j_1,\dots,j_n\}$ with $\sigma_S^{\varepsilon,T}(\Phi_j,\Psi_j) = \sigma_S^{\varepsilon,T}(\Phi_k,\Psi_k)$, for all $k \in \{j_1,\dots,j_n\}$. If $\varepsilon \leq \min\{R_{Fd}(\Phi_i,\Phi_j) \mid i \in \{1,\dots,m\}\}$, then the N_ε-efficiency of $DA_T(S)$ is*

$$\eta^{N_\varepsilon}(DA_T(S)) = \sigma_S^{\varepsilon,T}(\Phi_j,\Psi_j)$$

Proof. We suppose that $\varepsilon \leq \min\{R_{Fd}(\Phi_i,\Phi_j) \mid i \in \{1,\dots,m\}\}$. By Definition 6 it is obtained that $[\Phi_j]_\varepsilon = \{\Phi_1,\dots,\Phi_m\}$. Taking into account that the ε-T-strength of $\Phi_j \to \Psi_j$ with respect to $Dec(S)$ is greater than the ε-T-strengths of the rest of rules of the algorithm we deduce that

$$\Phi_j \to \Psi_j \in B_{\Phi_i,\Psi_i} \quad \text{for all } i \in \{1,\dots,m\} \tag{1}$$

$$C_{\Phi_i,\Psi_i} = C_{\Phi_k,\Psi_k} \quad \text{for all } i,k \in \{1,\dots,m\} \tag{2}$$

$$D_{\Phi_1,\Psi_1} = C_{\Phi_1,\Psi_1} \tag{3}$$

$$D_{\Phi_k,\Psi_k} = \varnothing \quad \text{for all } k \in \{2,\dots,m\} \tag{4}$$

By Equations (1) and (2), and Definitions 15 and 17 we obtain that:

$$\sigma_S^{\varepsilon,T}(\Phi_j,\Psi_j) = \frac{\sum\limits_{\{\Phi''\to\Psi''\in DA_T(S) \mid \Phi''\in[\Phi_j]_\varepsilon, \Psi''\in[\Psi_j]_\varepsilon\}} supp_S^T(\Phi'',\Psi'')}{\sum\limits_{\Phi'''\to\Psi'''\in DA_T(S)} supp_S^T(\Phi''',\Psi''')} = \frac{\sum\limits_{\Phi''\to\Psi''\in C_{\Phi_1,\Psi_1}} supp_S^T(\Phi'',\Psi'')}{\sum\limits_{\Phi'''\to\Psi'''\in DA_T(S)} supp_S^T(\Phi''',\Psi''')} \tag{5}$$

Therefore, by Definition 17 and Equations (3), (4) and (5) we conclude that

$$\eta^{N_\varepsilon}(DA_T(S)) = \frac{\sum\limits_{i=1}^{m}\sum\limits_{\Phi''\to\Psi''\in D_{\Phi_i,\Psi_i}} supp_S^T(\Phi'',\Psi'')}{\sum\limits_{\Phi'''\to\Psi'''\in DA_T(S)} supp_S^T(\Phi''',\Psi''')} = \frac{\sum\limits_{\Phi''\to\Psi''\in D_{\Phi_1,\Psi_1}} supp_S^T(\Phi'',\Psi'')}{\sum\limits_{\Phi'''\to\Psi'''\in DA_T(S)} supp_S^T(\Phi''',\Psi''')}$$

Table 1
Tables associated with the decision table S (left), decision rules (center), T-support and 0.6-T-strength of the decision rules (right) given in Example 21.

	a_1	d
x_1	0.4	0.9
x_2	0.55	0.4
x_3	0.95	0.95
x_4	0.5	0
x_5	1	1
x_6	0	0.7

Rules	$\Phi \to \Psi$
r_1	$(a_1, 0.4) \to (d, 0.9)$
r_2	$(a_1, 0.55) \to (d, 0.4)$
r_3	$(a_1, 0.95) \to (d, 0.95)$
r_4	$(a_1, 0.5) \to (d, 0)$
r_5	$(a_1, 1) \to (d, 1)$
r_6	$(a_1, 0) \to (d, 0.7)$

Rule	$supp_S^T$	$\sigma_S^{0.6,T}$
r_1	3.05	0.33
r_2	3.4	0.33
r_3	2.95	0.34
r_4	2.05	0.33
r_5	2.75	0.34
r_6	2.4	0.33

$$= \frac{\sum_{\Phi'' \to \Psi'' \in C_{\Phi_1,\Psi_1}} supp_S^T(\Phi'', \Psi'')}{\sum_{\Phi''' \to \Psi''' \in DA_T(S)} supp_S^T(\Phi''', \Psi''')}$$

$$= \sigma_S^{\varepsilon,T}(\Phi_j, \Psi_j) \quad \square$$

Notice that, under the considered hypothesis for the threshold ε, we cannot discern the antecedent of the first decision rule of the algorithm with maximum ε-T-strength, from the antecedents of the rest of rules. As a result, this rule is the most representative independently on the analyzed antecedent, that is, it belongs to all the sets B_{Φ_i,Ψ_i}, which leads us to obtain a N_ε-efficiency equals to the maximum of all the ε-T-strengths.

Now, we provide a particular condition for the threshold ε in order to obtain a completely efficient $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm, that is, with N_ε-efficiency equals to 1.

Proposition 20. *Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table and $DA_T(S) = \{\Phi_i \to \Psi_i \mid i \in \{1, \dots, m\}\}$, with $m \geq 2$, a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm. If $\max\{R_{Fd}(\Phi_i, \Phi_j) \mid i, j \in \{1, \dots, m\}, i \neq j\} < \varepsilon$ then the N_ε-efficiency of $DA_T(S)$ is*

$$\eta^{N_\varepsilon}(DA_T(S)) = 1$$

Proof. Since $\max\{R_{Fd}(\Phi_i, \Phi_j) \mid i, j \in \{1, \dots, m\}, i \neq j\} < \varepsilon$ we obtain that $\left[\Phi_i\right]_\varepsilon = \{\Phi_i\}$ for all $i \in \{1, \dots, m\}$. Hence, by Definition 17, the next chain of equalities $B_{\Phi_i,\Psi_i} = C_{\Phi_i,\Psi_i} = D_{\Phi_i,\Psi_i} = \{\Phi_i \to \Psi_i\}$ holds, for all $i \in \{1, \dots, m\}$. Applying again Definition 17, we deduce that:

$$\eta^{N_\varepsilon}(DA_T(S)) = \frac{\sum_{i=1}^{m} \sum_{\Phi'' \to \Psi'' \in D_{\Phi_i,\Psi_i}} supp_S^T(\Phi'', \Psi'')}{\sum_{\Phi''' \to \Psi''' \in DA_T(S)} supp_S^T(\Phi''', \Psi''')} = \frac{\sum_{i=1}^{m} supp_S^T(\Phi_i, \Psi_i)}{\sum_{\Phi''' \to \Psi''' \in DA_T(S)} supp_S^T(\Phi''', \Psi''')} = 1 \quad \square$$

Notice that, all the antecedents have to be different from each other, otherwise the given hypothesis is not satisfied. Hence, if the considered threshold ε allows us to discern all the antecedents of the decision rules present in the algorithm, then we do not detect inconsistencies in it, because they take place when we cannot discern antecedents of some rules, but we can discern their consequents. As a result, the decision algorithm is completely efficient, that is, its N_ε-efficiency is 1.

The sets C_{Φ_i,Ψ_i} defined above consider a minimum, for all $i \in \{1, \dots, m\}$. This fact is necessary in order to establish a criteria when $\text{card}(B_{\Phi_i,\Psi_i}) > 1$, since different N_ε-efficiencies can be obtained for the same algorithm and threshold ε, depending on the order of the rules in that algorithm, as next example shows.

Example 21. Consider the decision table $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ represented in Table 1, where the set of objects is $U = \{x_1, x_2, x_3, x_4, x_5, x_6\}$, the set of attributes is $\mathcal{A}_d = \{a_1, d\}$, and $\mathcal{V}_{\mathcal{A}_d} = \{V_{a_1}, V_d\}$, with $V_{a_1} = V_d = [0, 1]$.

Thus, we define the set of decision rules $Dec(S) = \{r_1, r_2, r_3, r_4, r_5, r_6\}$ where each rule is induced by a different object of the decision table. Moreover, we consider the family of separable fuzzy tolerance relations $T = \{T_a : V_a \times V_a \to [0, 1] \mid a \in \mathcal{A}_d\}$, defined as $T_a(\bar{a}(x), v) = 1 - |\bar{a}(x) - v|$ for all $a \in \mathcal{A}_d$, $x \in U$ and $v \in [0, 1]$, and the threshold $\varepsilon = 0.6$. The given decision rules, their T-support and their 0.6-T-strength are shown in Table 1.

By using Definition 6, we compute the relationship between the antecedents and the consequents of decision rules of $Dec(S)$, in order to obtain the set of 0.6-consistent decision rules. The obtained results are shown in Table 2.

Table 2
Relationship between the antecedents and the consequents, respectively, of the decision rules of Example 21.

R_{Fd}	Φ_1	Φ_2	Φ_3	Φ_4	Φ_5	Φ_6
Φ_1	1	0.85	0.45	0.9	0.4	0.6
Φ_2	0.85	1	0.6	0.95	0.55	0.45
Φ_3	0.45	0.6	1	0.55	0.95	0.05
Φ_4	0.9	0.95	0.55	1	0.5	0.5
Φ_5	0.4	0.55	0.95	0.5	1	0
Φ_6	0.6	0.45	0.05	0.5	0	1

R_{Fd}	Ψ_1	Ψ_2	Ψ_3	Ψ_4	Ψ_5	Ψ_6
Ψ_1	1	0.5	0.95	0.1	0.9	0.8
Ψ_2	0.5	1	0.45	0.6	0.4	0.7
Ψ_3	0.95	0.45	1	0.05	0.95	0.75
Ψ_4	0.1	0.6	0.05	1	0	0.3
Ψ_5	0.9	0.4	0.95	0	1	0.7
Ψ_6	0.8	0.7	0.75	0.3	0.7	1

Table 3
Values and sets needed to compute the $N_{0.6}$-efficiency of $DA_T(S)$.

	$\eta^{N_{0.6}}_{\Phi_i}(DA_T(S))$	B_{Φ_i,Ψ_i}	C_{Φ_i,Ψ_i}	D_{Φ_i,Ψ_i}
$i=1$	0.33	$\{r_1,r_2,r_4,r_6\}$	$\{r_1,r_6\}$	$\{r_1,r_6\}$
$i=2$	0.34	$\{r_3\}$	$\{r_3,r_5\}$	$\{r_3,r_5\}$
$i=3$	0.34	$\{r_3,r_5\}$	$\{r_3,r_5\}$	$\varnothing$
$i=4$	0.33	$\{r_1,r_2,r_4\}$	$\{r_1,r_6\}$	$\varnothing$
$i=5$	0.34	$\{r_3,r_5\}$	$\{r_3,r_5\}$	$\varnothing$
$i=6$	0.33	$\{r_1,r_6\}$	$\{r_1,r_6\}$	$\varnothing$

Hence, from Definition 6 and Table 2, we compute the R_{Fd}-0.6-blocks:

$$[\Phi_1]_{0.6}=\{\Phi_1,\Phi_2,\Phi_4,\Phi_6\} \qquad [\Psi_1]_{0.6}=\{\Psi_1,\Psi_3,\Psi_5,\Psi_6\}$$
$$[\Phi_2]_{0.6}=\{\Phi_1,\Phi_2,\Phi_3,\Phi_4\} \qquad [\Psi_2]_{0.6}=\{\Psi_2,\Psi_4,\Psi_6\}$$
$$[\Phi_3]_{0.6}=\{\Phi_2,\Phi_3,\Phi_5\} \qquad [\Psi_3]_{0.6}=\{\Psi_1,\Psi_3,\Psi_5,\Psi_6\}$$
$$[\Phi_4]_{0.6}=\{\Phi_1,\Phi_2,\Phi_4\} \qquad [\Psi_4]_{0.6}=\{\Psi_2,\Psi_4\}$$
$$[\Phi_5]_{0.6}=\{\Phi_3,\Phi_5\} \qquad [\Psi_5]_{0.6}=\{\Psi_1,\Psi_3,\Psi_5,\Psi_6\}$$
$$[\Phi_6]_{0.6}=\{\Phi_1,\Phi_6\} \qquad [\Psi_6]_{0.6}=\{\Psi_1,\Psi_2,\Psi_3,\Psi_5,\Psi_6\}$$

Taking into account Definition 7 and the previous R_{Fd}-0.6-blocks, we have that the set of 0.6-consistent decision rules is:

$$Dec^+_{0.6}(S)=\{\Phi_5\to\Psi_5,\Phi_6\to\Psi_6\}$$

As a consequence, by using a similar approach to the one detailed in [9] and the Łukasiewicz fuzzy implication in the computation of the multi-adjoint fuzzy $\mathcal{A}$-positive region, we conclude that if $\alpha_1\geq 0.95$, $\alpha_2\geq 0.95$, $\alpha_3\leq 2.05$ and $\alpha_4\geq 0.5$, then $Dec(S)$ is a $(\alpha_1,\alpha_2,\alpha_3,\alpha_4)_{0.6}$-decision algorithm. From now on, let $DA_T(S)=\{r_1,r_2,r_3,r_4,r_5,r_6\}$.

Now, we compute the $N_{0.6}$-efficiency of $DA_T(S)$. Consider the decision rule $\Phi_1\to\Psi_1\in DA_T(S)$, from Definition 17 and Table 1, we deduce that:

$$\begin{aligned}
\eta^{N_{0.6}}_{\Phi_1}(DA_T(S)) &= \max\{\sigma^{0.6,T}_S(\Phi,\Psi)\mid \Phi\to\Psi\in DA_T(S),\Phi\in[\Phi_1]_{0.6}\}\\
&= \max\{\sigma^{0.6,T}_S(\Phi_1,\Psi_1),\sigma^{0.6,T}_S(\Phi_2,\Psi_2),\sigma^{0.6,T}_S(\Phi_4,\Psi_4),\sigma^{0.6,T}_S(\Phi_6,\Psi_6)\}\\
&= \max\{0.33,0.33,0.33,0.33\}\\
&= 0.33\\
B_{\Phi_1,\Psi_1} &= \{\Phi'\to\Psi'\in DA_T(S)\mid \sigma^{0.6,T}_S(\Phi',\Psi')=0.33,\Phi'\in[\Phi_1]_{0.6}\}\\
&= \{\Phi_1\to\Psi_1,\Phi_2\to\Psi_2,\Phi_4\to\Psi_4,\Phi_6\to\Psi_6\}\\
C_{\Phi_1,\Psi_1} &= \{\Phi''\to\Psi''\in DA_T(S)\mid \Phi''\in[\Phi_k]_{0.6},\Psi''\in[\Psi_k]_{0.6},k=\min\{l\in\{1,\dots,m\}\mid \Phi_l\to\Psi_l\in B_{\Phi_1,\Psi_1}\}\}\\
&= \{\Phi''\to\Psi''\in DA_T(S)\mid \Phi''\in[\Phi_1]_{0.6},\Psi''\in[\Psi_1]_{0.6}\}\\
&= \{\Phi_1\to\Psi_1,\Phi_6\to\Psi_6\}\\
D_{\Phi_1,\Psi_1} &= C_{\Phi_1,\Psi_1}\setminus\{\Phi''\to\Psi''\in D_{\Phi_j,\Psi_j}\mid j<1\}\\
&= C_{\Phi_1,\Psi_1}
\end{aligned}$$

Analogously, we obtain the values and sets related to the rest of rules. All the results are presented in Table 3.

Table 4
Values and sets needed to compute the $N_{0.6}$-efficiency of $DA_T^{\pi}(S)$.

	$\eta^{N_{0.6}}_{\Phi_{\pi(i)}}(DA_T^{\pi}(S))$	$B_{\Phi_{\pi(i)},\Psi_{\pi(i)}}$	$C_{\Phi_{\pi(i)},\Psi_{\pi(i)}}$	$D_{\Phi_{\pi(i)},\Psi_{\pi(i)}}$
$i=1$	0.34	$\{r_{\pi(3)}\}$	$\{r_{\pi(3)}, r_{\pi(5)}\}$	$\{r_{\pi(3)}, r_{\pi(5)}\}$
$i=2$	0.33	$\{r_{\pi(1)}, r_{\pi(2)}, r_{\pi(4)}, r_{\pi(6)}\}$	$\{r_{\pi(1)}, r_{\pi(4)}\}$	$\{r_{\pi(1)}, r_{\pi(4)}\}$
$i=3$	0.34	$\{r_{\pi(3)}, r_{\pi(5)}\}$	$\{r_{\pi(3)}, r_{\pi(5)}\}$	$\varnothing$
$i=4$	0.33	$\{r_{\pi(1)}, r_{\pi(2)}, r_{\pi(4)}\}$	$\{r_{\pi(1)}, r_{\pi(4)}\}$	$\varnothing$
$i=5$	0.34	$\{r_{\pi(3)}, r_{\pi(5)}\}$	$\{r_{\pi(3)}, r_{\pi(5)}\}$	$\varnothing$
$i=6$	0.33	$\{r_{\pi(2)}, r_{\pi(6)}\}$	$\{r_{\pi(2)}, r_{\pi(6)}\}$	$\{r_{\pi(2)}, r_{\pi(6)}\}$

Hence, by Definition 17 and Table 1, we obtain that the $N_{0.6}$-efficiency of $DA_T(S)$ is:

$$\begin{aligned}\eta^{N_{0.6}}(DA_T(S)) &= \frac{\sum_{i=1}^{6}\sum_{\Phi''\to\Psi''\in D_{\Phi_i,\Psi_i}} supp_S^T(\Phi'',\Psi'')}{\sum_{\Phi'''\to\Psi'''\in DA_T(S)} supp_S^T(\Phi''',\Psi''')} = \frac{supp_S^T(\Phi_1,\Psi_1)+supp_S^T(\Phi_6,\Psi_6)+supp_S^T(\Phi_3,\Psi_3)+supp_S^T(\Phi_5,\Psi_5)}{\sum_{\Phi'''\to\Psi'''\in DA_T(S)} supp_S^T(\Phi''',\Psi''')}\\ &= \frac{3.05+2.4+2.95+2.75}{3.05+3.4+2.95+2.05+2.75+2.4} = \frac{11.5}{16.6}\\ &= 0.69\end{aligned}$$

On the other hand, in order to show the importance of the order of the rules in this case, now we exchange the order of the decision rules r_1 and r_2, by means of the permutation $\pi : \{1,\dots,6\} \to \{1,\dots,6\}$ defined as $\pi(1)=2$, $\pi(2)=1$ and $\pi(j)=j$, for all $j \in \{3,\dots,6\}$. We obtain the following decision algorithm $DA_T^{\pi}(S)$ and the equivalences of the considered relevance indicators between the rules of both algorithms:

$$\begin{aligned}
&r_{\pi(1)} : (a_1, 0.55) \to (d, 0.4) && supp_S^T(\Phi_{\pi(1)},\Psi_{\pi(1)}) = supp_S^T(\Phi_2,\Psi_2)\\
&r_{\pi(2)} : (a_1, 0.4) \to (d, 0.9) && supp_S^T(\Phi_{\pi(2)},\Psi_{\pi(2)}) = supp_S^T(\Phi_1,\Psi_1)\\
&r_{\pi(3)} : (a_1, 0.95) \to (d, 0.95) && supp_S^T(\Phi_{\pi(k)},\Psi_{\pi(k)}) = supp_S^T(\Phi_k,\Psi_k) \text{ for } k \in \{3,\dots,6\}\\
&r_{\pi(4)} : (a_1, 0.5) \to (d, 0) && \sigma_S^{0.6,T}(\Phi_{\pi(1)},\Psi_{\pi(1)}) = \sigma_S^{0.6,T}(\Phi_2,\Psi_2)\\
&r_{\pi(5)} : (a_1, 1) \to (d, 1) && \sigma_S^{0.6,T}(\Phi_{\pi(2)},\Psi_{\pi(2)}) = \sigma_S^{0.6,T}(\Phi_1,\Psi_1)\\
&r_{\pi(6)} : (a_1, 0) \to (d, 0.7) && \sigma_S^{0.6,T}(\Phi_{\pi(k)},\Psi_{\pi(k)}) = \sigma_S^{0.6,T}(\Phi_k,\Psi_k) \text{ for } k \in \{3,\dots,6\}
\end{aligned}$$

With respect to the R_{Fd}-0.6-blocks of the antecedents and consequents we deduce that:

$$\begin{aligned}
&[\Phi_{\pi(1)}]_{0.6} = \{\Phi_{\pi(1)},\Phi_{\pi(2)},\Phi_{\pi(3)},\Phi_{\pi(4)}\} && [\Psi_{\pi(1)}]_{0.6} = \{\Psi_{\pi(1)},\Psi_{\pi(4)},\Psi_{\pi(6)}\}\\
&[\Phi_{\pi(2)}]_{0.6} = \{\Phi_{\pi(1)},\Phi_{\pi(2)},\Phi_{\pi(4)},\Phi_{\pi(6)}\} && [\Psi_{\pi(2)}]_{0.6} = \{\Psi_{\pi(2)},\Psi_{\pi(3)},\Psi_{\pi(5)},\Psi_{\pi(6)}\}\\
&[\Phi_{\pi(3)}]_{0.6} = \{\Phi_{\pi(1)},\Phi_{\pi(3)},\Phi_{\pi(5)}\} && [\Psi_{\pi(3)}]_{0.6} = \{\Psi_{\pi(2)},\Psi_{\pi(3)},\Psi_{\pi(5)},\Psi_{\pi(6)}\}\\
&[\Phi_{\pi(4)}]_{0.6} = \{\Phi_{\pi(1)},\Phi_{\pi(2)},\Phi_{\pi(4)}\} && [\Psi_{\pi(4)}]_{0.6} = \{\Psi_{\pi(1)},\Psi_{\pi(4)}\}\\
&[\Phi_{\pi(5)}]_{0.6} = \{\Phi_{\pi(3)},\Phi_{\pi(5)}\} && [\Psi_{\pi(5)}]_{0.6} = \{\Psi_{\pi(2)},\Psi_{\pi(3)},\Psi_{\pi(5)},\Psi_{\pi(6)}\}\\
&[\Phi_{\pi(6)}]_{0.6} = \{\Phi_{\pi(2)},\Phi_{\pi(6)}\} && [\Psi_{\pi(6)}]_{0.6} = \{\Psi_{\pi(1)},\Psi_{\pi(2)},\Psi_{\pi(3)},\Psi_{\pi(5)},\Psi_{\pi(6)}\}
\end{aligned}$$

Hence, by Definition 17, we obtain the results given in Table 4.
In conclusion, applying Definition 17, the $N_{0.6}$-efficiency of $DA_T^{\pi}(S)$ is:

$$\eta^{N_{0.6}}(DA_T^{\pi}(S)) = \frac{\sum_{i=1}^{6}\sum_{\Phi''\to\Psi''\in D_{\Phi_{\pi(i)},\Psi_{\pi(i)}}} supp_S^T(\Phi'',\Psi'')}{\sum_{\Phi'''\to\Psi'''\in DA_T^{\pi}(S)} supp_S^T(\Phi''',\Psi''')} = 1$$

In this way, the consideration of different orders in the set of rules of a $(\alpha_1,\alpha_2,\alpha_3,\alpha_4)_\alpha$-decision algorithm can leads us to obtain different N_ε-efficiencies for the same threshold ε. □

Example 21 shows that different N_ε-efficiencies can be obtained for a given algorithm $DA_T(S)$. Although the user can solve this problem by choosing different condition attributes in the antecedents of the rules belonging to $DA_T(S)$, redefining the family of separable fuzzy tolerance relations T or modifying the threshold ε, it is convenient to establish a strict order in the set of decision rules

Table 5
Tables associated with S (left) and decision rules (right) given in Example 23.

	a_1	a_2	a_3	d
x_1	0.34	0.31	0.75	0
x_2	0.21	0.71	0.5	1
x_3	0.52	0.92	1	0
x_4	0.85	0.65	1	1
x_5	0.43	0.89	0.5	0
x_6	0.21	0.47	0.25	1
x_7	0.09	0.93	0.25	0

Rules	$\Phi \to \Psi$
r_1	$(a_1, 0.34) \wedge (a_2, 0.31) \wedge (a_3, 0.75) \to (d, 0)$
r_2	$(a_1, 0.21) \wedge (a_2, 0.71) \wedge (a_3, 0.5) \to (d, 1)$
r_3	$(a_1, 0.52) \wedge (a_2, 0.92) \wedge (a_3, 1) \to (d, 0)$
r_4	$(a_1, 0.85) \wedge (a_2, 0.65) \wedge (a_3, 1) \to (d, 1)$
r_5	$(a_1, 0.43) \wedge (a_2, 0.89) \wedge (a_3, 0.5) \to (d, 0)$
r_6	$(a_1, 0.21) \wedge (a_2, 0.47) \wedge (a_3, 0.25) \to (d, 1)$
r_7	$(a_1, 0.09) \wedge (a_2, 0.93) \wedge (a_3, 0.25) \to (d, 0)$

of $DA_T(S)$ to avoid these situations. This strict order consists in an irreflexive, asymmetric and transitive binary relation presented in the following definition.

Definition 22. Let $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ be a decision table, $DA_T(S)$ be a $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm and $\varepsilon \in [0,1]$. We say that $<_r$ is a *strict ordering on* $DA_T(S)$ if one of the following conditions is satisfied, for each pair of rules $\Phi \to \Psi, \Phi' \to \Psi' \in DA_T(S)$:

1. $\text{card}([\Phi]_\varepsilon) > \text{card}(\left[\Phi'\right]_\varepsilon)$.
2. $\text{card}([\Phi]_\varepsilon) = \text{card}(\left[\Phi'\right]_\varepsilon)$ and $\sigma_S^{\varepsilon,T}(\Phi, \Psi) > \sigma_S^{\varepsilon,T}(\Phi', \Psi')$.
3. $\text{card}([\Phi]_\varepsilon) = \text{card}(\left[\Phi'\right]_\varepsilon)$, $\sigma_S^{\varepsilon,T}(\Phi, \Psi) = \sigma_S^{\varepsilon,T}(\Phi', \Psi')$ and $supp_S^T(\Phi, \Psi) > supp_S^T(\Phi', \Psi')$.

In this case, we write $\Phi \to \Psi <_r \Phi' \to \Psi'$. Otherwise, we will say that ε *degenerates* $DA_T(S)$.

Notice that, this definition allows the user to manage the most important information in suitable way, since the most representative rules are given at the beginning of the algorithm, that is, the rules with more objects satisfying the antecedents, those with greater ε-T-strength or those with greater T-support. From now on, if we compute the N_ε-efficiency of a decision algorithm by Definition 17, then we have to present the decision rules of the algorithm according to the order determined by $<_r$, from the least rule to the greatest one.

It is convenient to consider decision algorithms that are not degenerated by a threshold ε to compute the N_ε-efficiency. Otherwise, we can obtain two different N_ε-efficiencies from the same algorithm. However, it is very strange this situation, since the algorithm must contain two rules satisfying the three previous conditions, and even in this case, there are situations where the N_ε-efficiency does not depend on the corresponding order. Indeed, given a decision algorithm $DA_T(S)$ and two rules $\Phi \to \Psi, \Phi' \to \Psi' \in DA_T(S)$, with $\Phi \to \Psi \nless_r \Phi' \to \Psi'$ nor $\Phi' \to \Psi' \nless_r \Phi \to \Psi$ such that one of the following conditions is verified:

1. $[\Phi]_\varepsilon \cap \left[\Phi'\right]_\varepsilon = \varnothing$
2. For each $\Phi'' \to \Psi'' \in DA_T(S)$ such that $\Phi, \Phi' \in \left[\Phi''\right]_\varepsilon$, it is obtained

$$\sigma_S^{\varepsilon,T}(\Phi, \Psi) = \sigma_S^{\varepsilon,T}(\Phi', \Psi') < \eta_{\Phi''}^{N_\varepsilon}(DA_T(S))$$

Then, we can order this pair of rules in the algorithm $DA_T(S)$ indifferently, since we will obtain the same N_ε-efficiency in both cases. From Item 1, we deduce that, if $\Phi \in \left[\Phi''\right]_\varepsilon$ then $\Phi' \notin \left[\Phi''\right]_\varepsilon$, and if $\Phi' \in \left[\Phi'''\right]_\varepsilon$ then $\Phi \notin \left[\Phi'''\right]_\varepsilon$ for all $\Phi'' \to \Psi'', \Phi''' \to \Psi''' \in DA_T(S)$. As a result, Φ, Φ' do not belong to the same set B_{Φ_i,Ψ_i} for any $\Phi_i \to \Psi_i \in DA_T(S)$, so we do not have to choose between the rules $\Phi \to \Psi$ and $\Phi' \to \Psi'$ in any C_{Φ_i,Ψ_i}. The other condition which determines the sets B_{Φ_i,Ψ_i} is that the ε-T-strength of the rules belonging to this set is the maximum among the rules whose antecedents are R_{Fd}-ε-related to Φ_i. This fact does not happen with the rules $\Phi \to \Psi$ and $\Phi' \to \Psi'$ for all Φ'' such that $\Phi, \Phi' \in \left[\Phi''\right]_\varepsilon$, by Item 2. Hence, once again, Φ, Φ' do not belong to the same set B_{Φ_i,Ψ_i} for any $\Phi_i \to \Psi_i \in DA_T(S)$.

Finally, next example is devoted to analyzing the N_ε-efficiency of the $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm given in [9], taking advantage of Definition 17 and Propositions 19 and 20.

Example 23. Consider the decision table $S = (U, \mathcal{A}_d, \mathcal{V}_{\mathcal{A}_d}, \overline{\mathcal{A}_d})$ represented in Table 5, where the set of objects is $U = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$, the set of attributes is $\mathcal{A}_d = \{a_1, a_2, a_3, d\}$ and $\mathcal{V}_{\mathcal{A}_d} = \{V_a \mid a \in \mathcal{A}_d\}$ with $V_a = [0,1]$, for all $a \in \mathcal{A}_d$. Taking into account all the condition attributes of S, we define the decision rules $r_i : \Phi_i \to \Psi_i$, for all $i \in \{1, \dots, 7\}$, displayed also in Table 5.

Taking $\alpha = 0.75$, $\alpha_1 \geq 0.78$, $\alpha_2 \geq 0$, $\alpha_3 \leq 1.61$ and $\alpha_4 \geq 0.39$, we obtain that $DA_T(S) = \{r_1, r_2, r_3, r_4, r_5, r_6, r_7\}$ is a complete $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)_\alpha$-decision algorithm. See [9] for more details.

From now on, we consider the family of separable fuzzy tolerance relations $T = \{T_a : V_a \times V_a \to [0,1] \mid a \in \mathcal{A}_d\}$, defined as $T_a(v, w) = 1 - |v - w|$ for all $a \in \mathcal{A}_d$ and $v, w \in V_a$. First of all, the antecedents of the rules are compared applying Definition 3, obtaining the results shown in Table 6.

Table 6
Relation between each pair of antecedents of decision rules of Example 23.

R_{Fd}	Φ_1	Φ_2	Φ_3	Φ_4	Φ_5	Φ_6	Φ_7
Φ_1	1	0.6	0.39	0.49	0.42	0.5	0.38
Φ_2	0.6	1	0.5	0.36	0.78	0.75	0.75
Φ_3	0.39	0.5	1	0.67	0.5	0.25	0.25
Φ_4	0.49	0.36	0.67	1	0.5	0.25	0.24
Φ_5	0.42	0.78	0.5	0.5	1	0.58	0.66
Φ_6	0.5	0.75	0.25	0.25	0.58	1	0.54
Φ_7	0.38	0.75	0.25	0.24	0.66	0.54	1

Table 7
Some important relevance indicators of the decision rules of Example 23.

Rule	$supp_S^T$	$\sigma_S^{0.33,T}$	$\sigma_S^{0.65,T}$	$\sigma_S^{0.67,T}$
r_1	2.19	0.62	0.15	0.15
r_2	2.11	0.38	0.28	0.28
r_3	2.14	0.46	0.14	0.14
r_4	1.61	0.25	0.11	0.11
r_5	2.58	0.62	0.33	0.17
r_6	2	0.28	0.28	0.28
r_7	2.29	0.47	0.33	0.15

Taking into account that the attribute d is Boolean, we deduce that:

$$R_{Fd}(\Psi_i,\Psi_j)=\begin{cases}1 & \text{if } i,j \text{ are both even or odd}\\ 0 & \text{otherwise}\end{cases}$$

Now, we compute the T-support and some ε-T-strengths by changing the threshold ε for all decision rules in $DA_T(S)$, which will be useful for the rest of the example. In particular, we consider the thresholds $\varepsilon=0.33$, $\varepsilon=0.65$ and $\varepsilon=0.67$. The obtained results are collected in Table 7.

From Table 7 we deduce that it does not exist $\varepsilon\in[0,1]$ that degenerates this decision algorithm, as all the decision rules have a different T-support. Now, we compute the N_ε-efficiency of $DA_T(S)$ by using Definition 17, Propositions 19 and 20, and considering the previous thresholds ε. We will begin by applying Propositions 19 and 20 to easily obtain some N_ε-efficiencies, and finally we will study a case in which these results cannot be applied.

First of all, in order to apply Proposition 19, it is necessary that the first decision rule $\Phi_j\to\Psi_j\in DA_T(S)$ with maximum ε-T-strength with respect to $DA_T(S)$ satisfies that $\varepsilon\le\min\{R_{Fd}(\Phi_i,\Phi_j)\mid i\in\{1,\dots,7\}\}$. According to Tables 6 and 7, we have that the decision rule r_1 satisfies this condition with the threshold $\varepsilon=0.33$. As a result, the $N_{0.33}$-efficiency of $DA_T(S)$ is:

$$\eta^{N_{0.33}}(DA_T^{\pi}(S))=\sigma_S^{0.33,T}(\Phi_1,\Psi_1)=0.62$$

We have an expected efficiency since all the antecedents are in the same block, and only two different rules have been considered. Notice that, the decision rules whose consequent is $\Psi=(d,1)$, which are r_2, r_4 and r_6, are not involved in the computation of the 0.33-T-strength of r_1, since their consequents are not related to $\Psi_1=(d,0)$. Therefore, the obtained value of efficiency is clearly associated with the representativeness of the decision $(d,0)$ in the decision algorithm.

On the other hand, we will now apply Proposition 20, which is focused on the relationship between the antecedents of the rules. By Table 6, we know that:

$$\max\{R_{Fd}(\Phi_i,\Phi_j)\mid i,j\in\{1,\dots,7\}, i\neq j\}=R_{Fd}(\Phi_2,\Phi_5)=0.78$$

As a result, taking $\varepsilon>0.78$ we deduce that the N_ε-efficiency of the $DA_T(S)$ is:

$$\eta^{N_\varepsilon}(DA_T(S))=1$$

Hence, $DA_T(S)$ is totally efficient given the threshold $\varepsilon>0.78$, since we can discern all the antecedents of the rules.

Finally, we compute the $N_{0.65}$-efficiency of the $(\alpha_1,\alpha_2,\alpha_3,\alpha_4)_{0.75}$- decision algorithm $DA_T(S)$. Notice that, the threshold $\varepsilon=0.65$ does not satisfy any property of Propositions 19 and 20. Hence, this case has to be computed by using Definition 17, so it is necessary to order the decision rules by Definition 22.

Table 8
Values and sets needed to compute the $N_{0.65}$-efficiency of $DA_T^{\pi}(S)$.

	$\eta^{N_{0.65}}_{\Phi_{\pi(i)}}(DA_T^{\pi}(S))$	$B_{\Phi_{\pi(i)},\Psi_{\pi(i)}}$	$C_{\Phi_{\pi(i)},\Psi_{\pi(i)}}$	$D_{\Phi_{\pi(i)},\Psi_{\pi(i)}}$
$i=1$	0.33	$\{r_{\pi(2)}, r_{\pi(3)}\}$	$\{r_{\pi(2)}, r_{\pi(3)}\}$	$\{r_{\pi(2)}, r_{\pi(3)}\}$
$i=2$	0.33	$\{r_{\pi(2)}, r_{\pi(3)}\}$	$\{r_{\pi(2)}, r_{\pi(3)}\}$	$\varnothing$
$i=3$	0.33	$\{r_{\pi(2)}, r_{\pi(3)}\}$	$\{r_{\pi(2)}, r_{\pi(3)}\}$	$\varnothing$
$i=4$	0.28	$\{r_{\pi(1)}, r_{\pi(4)}\}$	$\{r_{\pi(1)}, r_{\pi(4)}\}$	$\{r_{\pi(1)}, r_{\pi(4)}\}$
$i=5$	0.14	$\{r_{\pi(5)}\}$	$\{r_{\pi(5)}\}$	$\{r_{\pi(5)}\}$
$i=6$	0.14	$\{r_{\pi(5)}\}$	$\{r_{\pi(5)}\}$	$\varnothing$
$i=7$	0.15	$\{r_{\pi(7)}\}$	$\{r_{\pi(7)}\}$	$\{r_{\pi(7)}\}$

We begin by computing the R_{Fd}-0.65-blocks of all the antecedents. By Definition 6 and Table 6 we deduce that:

$$[\Phi_1]_{0.65} = \{\Phi_1\} \qquad [\Phi_4]_{0.65} = \{\Phi_3, \Phi_4\} \qquad [\Phi_6]_{0.65} = \{\Phi_2, \Phi_6\}$$
$$[\Phi_2]_{0.65} = \{\Phi_2, \Phi_5, \Phi_6, \Phi_7\} \qquad [\Phi_5]_{0.65} = \{\Phi_2, \Phi_5, \Phi_7\} \qquad [\Phi_7]_{0.65} = \{\Phi_2, \Phi_5, \Phi_7\}$$
$$[\Phi_3]_{0.65} = \{\Phi_3, \Phi_4\}$$

Now, taking into account these blocks and Table 7, we can order the rules in $DA_T(S)$ by Definition 22, obtaining that:

$$r_2 <_r r_5 <_r r_7 <_r r_6 <_r r_3 <_r r_4 <_r r_1$$

Hence, we define the permutation $\pi : \{1, \dots, 7\} \to \{1, \dots, 7\}$ given as $\pi(1) = 2$, $\pi(2) = 5$, $\pi(3) = 7$, $\pi(4) = 6$, $\pi(5) = 3$, $\pi(6) = 4$ and $\pi(7) = 1$, which leads us to the decision algorithm $DA_T^{\pi}(S)$:

$$r_{\pi(1)} : (a_1, 0.21) \wedge (a_2, 0.71) \wedge (a_3, 0.5) \to (d, 1)$$
$$r_{\pi(2)} : (a_1, 0.43) \wedge (a_2, 0.89) \wedge (a_3, 0.5) \to (d, 0)$$
$$r_{\pi(3)} : (a_1, 0.09) \wedge (a_2, 0.93) \wedge (a_3, 0.25) \to (d, 0)$$
$$r_{\pi(4)} : (a_1, 0.21) \wedge (a_2, 0.47) \wedge (a_3, 0.25) \to (d, 1)$$
$$r_{\pi(5)} : (a_1, 0.52) \wedge (a_2, 0.92) \wedge (a_3, 1) \to (d, 0)$$
$$r_{\pi(6)} : (a_1, 0.85) \wedge (a_2, 0.65) \wedge (a_3, 1) \to (d, 1)$$
$$r_{\pi(7)} : (a_1, 0.34) \wedge (a_2, 0.31) \wedge (a_3, 0.75) \to (d, 0)$$

Next, we compute the $N_{0.65}$-efficiency of $DA_T^{\pi}(S)$. By Definitions 6, 15 and 17 we obtain the results given in Table 8. Finally, applying Definition 17, the $N_{0.65}$-efficiency of $DA_T^{\pi}(S)$ is:

$$\begin{aligned}
\eta^{N_{0.65}}(DA_T^{\pi}(S)) &= \frac{\displaystyle\sum_{i=1}^{7} \sum_{\Phi'' \to \Psi'' \in D_{\Phi_{\pi(i)},\Psi_{\pi(i)}}} supp_S^T(\Phi'', \Psi'')}{\displaystyle\sum_{\Phi''' \to \Psi''' \in DA_T^{\pi}(S)} supp_S^T(\Phi''', \Psi''')} \\
&= \frac{1}{\displaystyle\sum_{\Phi''' \to \Psi''' \in DA_T^{\pi}(S)} supp_S^T(\Phi''', \Psi''')} \big(supp_S^T(\Phi_{\pi(2)}, \Psi_{\pi(2)}) + supp_S^T(\Phi_{\pi(3)}, \Psi_{\pi(3)}) + supp_S^T(\Phi_{\pi(1)}, \Psi_{\pi(1)}) \\
&\quad + supp_S^T(\Phi_{\pi(4)}, \Psi_{\pi(4)}) + supp_S^T(\Phi_{\pi(5)}, \Psi_{\pi(5)}) + supp_S^T(\Phi_{\pi(7)}, \Psi_{\pi(7)}) \big) \\
&= 0.89
\end{aligned}$$

Therefore, given the threshold $\varepsilon = 0.65$, the efficiency of the algorithm $DA_T^{\pi}(S)$ is equal to 0.89. This value is very high, taking into account that we have considered a low threshold. Consequently, given the threshold $\varepsilon = 0.65$, the algorithm $DA_T^{\pi}(S)$ reduces the inconsistency present in the decision table.

Although in these cases the N_ε-efficiency has been increasing in ε, excepting the N_0-efficiency which is always 1, notice that this property does not hold in general. In order to illustrate that, we will compute the $N_{0.67}$-efficiency of $DA_T(S)$.

The R_{Fd}-0.67-blocks of the antecedents or the rules of $DA_T(S)$ are very similar to the previous R_{Fd}-0.65-blocks, obtaining that:

$$[\Phi_k]_{0.67} = [\Phi_k]_{0.65} \qquad \text{for } k \in \{1, 2, 3, 4, 6\}$$
$$[\Phi_5]_{0.67} = \{\Phi_2, \Phi_5\}$$
$$[\Phi_7]_{0.67} = \{\Phi_2, \Phi_7\}$$

Table 9
Values and sets needed to compute the $N_{0.67}$-efficiency of $DA_T^{\pi}(S)$.

	$\eta^{N_{0.65}}_{\Phi_{\pi(i)}}(DA_T^{\pi}(S))$	$B_{\Phi_{\pi(i)},\Psi_{\pi(i)}}$	$C_{\Phi_{\pi(i)},\Psi_{\pi(i)}}$	$D_{\Phi_{\pi(i)},\Psi_{\pi(i)}}$
$i=1$	0.28	$\{r_{\pi(1)},r_{\pi(2)}\}$	$\{r_{\pi(1)},r_{\pi(2)}\}$	$\{r_{\pi(1)},r_{\pi(2)}\}$
$i=2$	0.28	$\{r_{\pi(1)},r_{\pi(2)}\}$	$\{r_{\pi(1)},r_{\pi(2)}\}$	$\varnothing$
$i=3$	0.28	$\{r_{\pi(1)}\}$	$\{r_{\pi(1)},r_{\pi(2)}\}$	$\varnothing$
$i=4$	0.28	$\{r_{\pi(1)}\}$	$\{r_{\pi(1)},r_{\pi(2)}\}$	$\varnothing$
$i=5$	0.14	$\{r_{\pi(5)}\}$	$\{r_{\pi(5)}\}$	$\{r_{\pi(5)}\}$
$i=6$	0.14	$\{r_{\pi(5)}\}$	$\{r_{\pi(5)}\}$	$\varnothing$
$i=7$	0.15	$\{r_{\pi(7)}\}$	$\{r_{\pi(7)}\}$	$\{r_{\pi(7)}\}$

As in the previous studies, the decision rules are ordered by Definition 22. Considering the permutation $\pi:\{1,\dots,7\}\to\{1,\dots,7\}$ defined as $\pi(1)=2$, $\pi(2)=6$, $\pi(3)=5$, $\pi(4)=7$, $\pi(5)=3$, $\pi(6)=4$ and $\pi(7)=1$, we obtain the following decision algorithm $DA_T^{\pi}(S)$:

$$r_{\pi(1)}:(a_1,0.21)\wedge(a_2,0.71)\wedge(a_3,0.5)\to(d,1)$$

$$r_{\pi(2)}:(a_1,0.21)\wedge(a_2,0.47)\wedge(a_3,0.25)\to(d,1)$$

$$r_{\pi(3)}:(a_1,0.43)\wedge(a_2,0.89)\wedge(a_3,0.5)\to(d,0)$$

$$r_{\pi(4)}:(a_1,0.09)\wedge(a_2,0.93)\wedge(a_3,0.25)\to(d,0)$$

$$r_{\pi(5)}:(a_1,0.52)\wedge(a_2,0.92)\wedge(a_3,1)\to(d,0)$$

$$r_{\pi(6)}:(a_1,0.85)\wedge(a_2,0.65)\wedge(a_3,1)\to(d,1)$$

$$r_{\pi(7)}:(a_1,0.34)\wedge(a_2,0.31)\wedge(a_3,0.75)\to(d,0)$$

Now, the $N_{0.67}$-efficiency of $DA_T^{\pi}(S)$ is computed. By Definitions 6, 15 and 17, we obtain the values and sets given in Table 9. Finally, by Definition 17, the $N_{0.67}$-efficiency of $DA_T^{\pi}(S)$ is:

$$\begin{aligned}\eta^{N_{0.67}}(DA_T^{\pi}(S)) &= \frac{\displaystyle\sum_{i=1}^{7}\sum_{\Phi''\to\Psi''\in D_{\Phi_{\pi(i)},\Psi_{\pi(i)}}} supp_S^T(\Phi'',\Psi'')}{\displaystyle\sum_{\Phi'''\to\Psi'''\in DA_T^{\pi}(S)} supp_S^T(\Phi''',\Psi''')}\\ &= \frac{1}{\displaystyle\sum_{\Phi'''\to\Psi'''\in DA_T^{\pi}(S)} supp_S^T(\Phi''',\Psi''')}\big(supp_S^T(\Phi_{\pi(1)},\Psi_{\pi(1)})+supp_S^T(\Phi_{\pi(2)},\Psi_{\pi(2)})\\ &\quad+supp_S^T(\Phi_{\pi(5)},\Psi_{\pi(5)})+supp_S^T(\Phi_{\pi(7)},\Psi_{\pi(7)})\big)\\ &=0.57\end{aligned}$$

Therefore, by considering a higher threshold, we have obtained a less efficient algorithm. This fact is due to the decision rules r_5 and r_7, which have the greatest T-support, are now less representative in S, since by Table 7, we have that:

$$\sigma_S^{0.65,T}(\Phi_5,\Psi_5)=\sigma_S^{0.65,T}(\Phi_7,\Psi_7)>\sigma_S^{0.67,T}(\Phi_5,\Psi_5)>\sigma_S^{0.67,T}(\Phi_7,\Psi_7)$$

This loss of representativeness in the table is caused by the changes occurred in the blocks of the antecedents, which are

$$[\Phi_5]_{0.65}=[\Phi_7]_{0.65}=\{\Phi_2,\Phi_5,\Phi_7\}$$

$$[\Phi_5]_{0.67}=\{\Phi_2,\Phi_5\}$$

$$[\Phi_7]_{0.67}=\{\Phi_2,\Phi_7\}$$

As a result, the rules r_5 and r_7 are not considered in the computation of the $N_{0.67}$-efficiency, since $\Phi_2\in[\Phi_5]_{0.67},[\Phi_7]_{0.67}$ and $\sigma_S^{0.67,T}(\Phi_2,\Psi_2)>\sigma_S^{0.67,T}(\Phi_5,\Psi_5)>\sigma_S^{0.67,T}(\Phi_7,\Psi_7)$. Nevertheless, these three decision rules are taken into account in the computation of the $N_{0.65}$-efficiency because $\sigma_S^{0.65,T}(\Phi_5,\Psi_5)=\sigma_S^{0.65,T}(\Phi_7,\Psi_7)>\sigma_S^{0.65,T}(\Phi_2,\Psi_2)$ and $B_{\Phi_6,\Phi_6}=\{r_2\}$, which is the reason for the existing difference between both N_ε-efficiencies. □

In conclusion, we have been able to extract relevant conclusions in Example 23 thanks to the introduction of Definition 17. In this way, we deduce that Definition 17 is especially useful to analyze $(\alpha_1,\alpha_2,\alpha_3,\alpha_4)_\alpha$-decision algorithms.

4. Conclusions and future work

This paper has presented two different notions for studying the efficiency of decision algorithms in fuzzy rough set theory, that is, ε-efficiency and N_ε-efficiency. We have shown that both notions of efficiency generalize the classical one, belonging the N_ε-efficiency to the unit interval. This fact is important because greatly helps to give an interpretation of the obtained value, which shows, for instance, the classifier behavior of the considered decision algorithm and how consistent is the original dataset.

We have also introduced the notion of ε-T-strength of a decision rule, which is a generalization of the strength of decision rules to the fuzzy framework. The aforementioned notion has played a key role in the definition of the N_ε-efficiency. Different properties and examples have been explained in order to illustrate the contribution and its relevance in the data analysis.

As future work, we will continue studying both new notions of efficiency to compare the efficiency of different decision algorithms. This fact will be of great importance for a better understanding about the attribute reduction and object classification in the fuzzy framework. On the other hand, we are interested in analyzing the relationship between the consistency of a given decision table and the efficiency of the decision algorithm considered to study such decision table.

CRediT authorship contribution statement

Fernando Chacón-Gómez: Visualization, Investigation, Writing – original draft, Methodology, Supervision, Conceptualization, Validation, Formal analysis, Writing – review & editing, Resources. **M. Eugenia Cornejo:** Supervision, Funding acquisition, Validation, Investigation, Writing – original draft, Project administration, Conceptualization, Writing – review & editing, Resources, Formal analysis, Visualization, Methodology. **Jesús Medina:** Writing – original draft, Project administration, Conceptualization, Writing – review & editing, Resources, Formal analysis, Validation, Investigation, Visualization, Methodology, Supervision, Funding acquisition.

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.

References

[1] M.J. Benítez-Caballero, J. Medina, Characterizing clarify and reduce method by means of left-sided formal concept analysis, Stud. Comput. Intell. X (2023) 1–6.
[2] M.J. Benítez-Caballero, J. Medina, E. Ramírez-Poussa, D. Ślęzak, Bireducts with tolerance relations, Inf. Sci. 435 (2018) 26–39.
[3] M.J. Benítez-Caballero, J. Medina-Moreno, E. Ramírez-Poussa, Bireducts in formal concept analysis, Springer International Publishing, Cham, 2020, pp. 191–198.
[4] F. Chacón-Gómez, M.E. Cornejo, J. Medina, Relating decision rules and attribute implications 3308 (2022) 69–78.
[5] F. Chacón-Gómez, M.E. Cornejo, J. Medina, Decision making in fuzzy rough set theory, Mathematics 11 (19) (2023).
[6] F. Chacón-Gómez, M.E. Cornejo, J. Medina, E. Ramírez-Poussa, Fuzzy rough set decision algorithms, Commun. Comput. Inf. Sci. 1601 (2022) 63–76.
[7] F. Chacón-Gómez, M.E. Cornejo, J. Medina, E. Ramírez-Poussa, Value reducts and bireducts: a comparative study, Math. Methods Appl. Sci. 46 (2) (2023) 1631–1650.
[8] F. Chacón-Gómez, M.E. Cornejo, J. Medina, E. Ramírez-Poussa, Efficiency of fuzzy rough set decision algorithms, Stud. Comput. Intell. 1127 (2024) 17–24.
[9] F. Chacón-Gómez, M.E. Cornejo, J. Medina, E. Ramírez-Poussa, Rough set decision algorithms for modeling with uncertainty, J. Comput. Appl. Math. 437 (2024) 115413.
[10] C. Cornelis, J. Medina, N. Verbiest, Multi-adjoint fuzzy rough sets: definition, properties and attribute selection, Int. J. Approx. Reason. 55 (2014) 412–426.
[11] A. De Luca, S. Termini, A definition of a nonprobabilistic entropy in the setting of fuzzy sets theory, Inf. Control 20 (4) (1972) 301–312.
[12] M. Grzegorowski, D. Ślęzak, On resilient feature selection: computational foundations of r-c-reducts, Inf. Sci. 499 (2019) 25–44.
[13] W. Huang, Y. She, X. He, W. Ding, Fuzzy rough sets-based incremental feature selection for hierarchical classification, IEEE Trans. Fuzzy Syst. 31 (10) (2023) 3721–3733.
[14] F. Jiang, X. Yu, D. Gong, J. Du, A random approximate reduct-based ensemble learning approach and its application in software defect prediction, Inf. Sci. 609 (2022) 1147–1168.
[15] R. Johnston, Regionalization and classification, in: K. Kempf-Leonard (Ed.), Encyclopedia of Social Measurement, Elsevier, New York, 2005, pp. 337–350.
[16] J. Medina, Multi-adjoint property-oriented and object-oriented concept lattices, Inf. Sci. 190 (2012) 95–106.
[17] R. Nisbet, G. Miner, K. Yale, Classification, in: R. Nisbet, G. Miner, K. Yale (Eds.), Handbook of Statistical Analysis and Data Mining Applications, second edition, Academic Press, Boston, 2018, pp. 169–186.
[18] Z. Pawlak, Information systems theoretical foundations, Inf. Syst. 6 (3) (1981) 205–218.
[19] Z. Pawlak, Rough sets, Int. J. Comput. Inf. Sci. 11 (1982) 341–356.
[20] Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning About Data, Kluwer Academic Publishers, Norwell, MA, USA, 1992.
[21] Z. Pawlak, Rough sets and decision algorithms, in: W. Ziarko, Y. Yao (Eds.), Rough Sets and Current Trends in Computing, Springer Berlin Heidelberg, 2001, pp. 30–45.
[22] B. Sang, L. Yang, H. Chen, W. Xu, X. Zhang, Fuzzy rough feature selection using a robust non-linear vague quantifier for ordinal classification, Expert Syst. Appl. 230 (2023) 120480.
[23] D. Ślęzak, A. Janusz, Ensembles of bireducts: towards robust classification and simple representation, in: T.-h. Kim, H. Adeli, D. Ślęzak, F. Sandnes, X. Song, K.-i. Chung, K. Arnett (Eds.), Future Generation Information Technology, in: Lecture Notes in Computer Science, vol. 7105, Springer Berlin Heidelberg, 2011, pp. 64–77.
[24] M. Szeląg, S. Greco, R. Słowiński, Similarity-based classification with dominance-based decision rules, in: V. Flores, F. Gomide, A. Janusz, C. Meneses, D. Miao, G. Peters, D. Ślęzak, G. Wang, R. Weber, Y. Yao (Eds.), Rough Sets, Springer International Publishing, Cham, 2016, pp. 355–364.
[25] T. Yang, J. Liang, Y. Pang, P. Xie, Y. Qian, R. Wang, An efficient feature selection algorithm based on the description vector and hypergraph, Inf. Sci. 629 (2023) 746–759.