Double Successive Rough Set Approximations

Alexa Gopaulsingh

Central European University
Budapest, Hungary
gopaulsingh_alexa@phd.ceu.edu

Abstract. We examine double successive approximations on a set, which we denote by $L_2 L_1$, $U_2 U_1$, $U_2 L_1$, $L_2 U_1$ where L_1, U_1 and L_2, U_2 are based on generally non-equivalent equivalence relations E_1 and E_2 respectively, on a finite non-empty set V. We consider the case of these operators being given fully defined on its powerset $\mathcal{P}(V)$. Then, we investigate if we can reconstruct the equivalence relations which they may be based on. Directly related to this, is the question of whether there are unique solutions for a given defined operator and the existence of conditions which may characterise this. We find and prove these characterising conditions that equivalence relation pairs should satisfy in order to generate unique such operators.

1 Successive Approximations

Double successive rough set approximations here, are considered using two, generally different equivalence relations. These are interesting because one can imagine a situation or model where sets/information to be approximated is input through two different approximations before returning the output. It is possible that for example heuristics in the brain can be modelled using such layered approximations. Decomposing successive approximations into constituent parts is somewhat analogous to decomposing a wave into sine and cosine parts using Fourier analysis.

In our case, we have two equivalence relations E_1 and E_2 on a set V with lower and upper approximations operators acting on its powerset $\mathcal{P}(V)$, denoted by L_1, U_1 and L_2, U_2 respectively. What if we knew the results of passing all the elements in $\mathcal{P}(V)$ through L_1 and then L_2, which we denote by $L_2 L_1$. Could we then reconstruct E_1 and E_2 from this information? In this paper, we will investigate this question and consider the four cases of being given a defined $L_2 L_1$, $U_2 U_1$, $U_2 L_1$, $L_2 U_1$ operators. We will find that two equivalence relations do not always produce unique such operators but that some pairs do. We find and characterise conditions which the pairs of equivalence relations must satisfy for them to produce a unique operator. Cattaneo and Ciucci found that preclusive relations are especially useful for rough approximations in information systems in [4]. For the $L_2 L_1$ case we will show that these conditions from a preclusive relation between pairs of equivalence relations on a set and so we can define a related notion of independence

from it. After this, we will find a more conceptual but equivalent version of the conditions of the uniqueness theorem. These conditions are more illuminating in that we can easier see why these conditions work while the conditions in the first version of the theorem are easier to use in practice. Lastly, we will consider the cases of the remaining operators, $U_2 U_1$, $U_2 L_1$ and $L_2 U_1$. We note that the $L_2 L_1$ and $U_2 U_1$ cases are dual to each other and similarly for the $U_2 L_1$ and $L_2 U_1$ cases.

Rough set theory has quite a large number of practical applications. This is due in part to the computation of reducts and decision rules for databases. Predictions can be made after the data is mined to extract decision rules of manageable size (i.e. attribute reduction). In this way, rough set theory can be used to make decisions using data in the absence of major prior assumptions as argued in more detail in [22]. Hence in retrospect, it is perhaps not so surprising that this leads to tremendous applications. Therefore, rough set analysis is added to the tools, which includes regression analysis and Bayes' Theorem, for pattern recognition and feature selection in data mining, see [19, 36, 10, 37, 29, 5, 26, 28]. The resulting applications include in medical databases [32, 31, 30, 8, 9, 12, 13], cognitive science [16, 14, 25, 17, 35], artificial intelligence and machine learning [15, 11, 7, 6, 33, 21, 27] and engineering [1, 2, 3, 24, 23]. Indeed in [34], Yao noted that there is currently an imbalance in the literature between the conceptual unfolding of rough set theory and its practical computational progress. He observed that the amount of computational literature currently far exceeds the amount of conceptual, theoretical literature. Moreover, he made the case that the field would prosper from a correction of this imbalance. To illustrate this, he began his recommendation in [34] by formulating a conceptual example of reducts that unifies three reduct definitions used in the literature which on the surface look different. We strongly agree that more efforts to find conceptual formulations of notions and results would increase the discovery of unifying notions. This would greatly aid the aim of making a cohesive and coherent map of the present mass of existing literature. In this direction, we have developed subsections 4.2 and 4.3 in section 4.

2   Basic Concepts of Rough Set Theory

We go over some basic notions and definitions which can be found in [18]. Let V be a set and E be an equivalence relation on V. Also, let the set of equivalence classes of E be denoted by V/E. If a set $X \subseteq V$, is equal to a union of some of the equivalence classes of E then it is called *E-exact*. Otherwise, X is called *E-inexact* or *E-rough* or simply *rough* when the equivalence relation under consideration is clear from the context. Inexact sets may be approximated by two exact sets, the lower and upper approximations as is respectively defined below:

$$\boldsymbol{l}_E(X) = \{x \in V \mid [x]_E \subseteq X\},$$

$$\boldsymbol{u}_E(X) = \{x \in V \mid [x]_E \cap X \neq \emptyset\}. \tag{1}$$

Equivalently, we may use a granule based definition instead of a pointwise based definition:

$$\boldsymbol{l}_E(X) = \bigcup \{Y \subseteq V/E \mid Y \subseteq X\},$$

$$\boldsymbol{u}_E(X) = \bigcup \{Y \subseteq V/E \mid Y \cap X \neq \emptyset\}. \tag{2}$$

The pair (V, E) is called an *approximation space*. It may be the case that several equivalence relations are considered over a set. Let $\mathcal{E}$ being a family of equivalence relations over a finite non-empty set V. The pair, $K = (V, \mathcal{E})$ is called *knowledge base*. If $\mathcal{P} \subseteq \mathcal{E}$, we recall that $\bigcap \mathcal{P}$ is alao

an equivalence relation. The intersection of all equivalence relations belonging to $\mathcal{P}$ is denoted by $IND(\mathcal{P}) = \bigcap \mathcal{P}$. This is called the *indiscernibility relation* over $\mathcal{P}$.

For two equivalence relations E_1 and E_2, we say that $E_1 \leq E_2$ iff $E_1 \subseteq E_2$. In this case we say that E_1 is *finer* than E_2 or that E_2 is *coarser* than E_1.

We recall from [20] some definitions about different types of roughly definable and undefinable sets. Let V be a set then for $X \subseteq V$:

(i) If $l_E(X) \neq \emptyset$ and $u_E(X) \neq V$, then X is called *roughly E-definable.*

(ii) If $l_E(X) = \emptyset$ and $u_E(X) \neq V$, then X is called *internally roughly E-undefinable.*

(iii) If $l_E(X) \neq \emptyset$ and $u_E(X) = V$, then X is called *externally roughly E-definable.*

(iv) If $l_E(X) = \emptyset$ and $u_E(X) = V$, then X is called *totally roughly E-definable.*

2.1 Properties Satisfied by Rough Sets

In [18], Pawlak enlists the following properties of lower and upper approximations. Let V be a non-empty finite set and $X, Y \subseteq V$. Then, the following holds:

1) $l_E(X) \subseteq X \subseteq u_E(X)$,

2) $l_E(\emptyset) = u_E(\emptyset) = \emptyset; \quad l_E(V) = u_E(V) = V$,

3) $u_E(X \cup Y) = u_E(X) \cup u_E(Y)$,

4) $l_E(X \cap Y) = l_E(X) \cap l_E(Y)$,

5) $X \subseteq Y \Rightarrow l_E(X) \subseteq l_E(Y)$,

6) $X \subseteq Y \Rightarrow u_E(X) \subseteq u_E(Y)$,

7) $l_E(X \cup Y) \supseteq l_E(X) \cup l_E(Y)$,

8) $u_E(X \cap Y) \supseteq u_E(X) \cap u_E(Y)$,

9) $l_E(-X) = -u_E(X)$,

10) $u_E(-X) = -l_E(X)$,

11) $l_E(l_E(X)) = u_E(l_E(X)) = l_E(X)$,

12) $u_E(u_E(X)) = l_E(u_E(X)) = u_E(X)$.

2.2 Dependencies in Knowledge Bases

A database can also be represented in the form of a matrix of *Objects* versus *Attributes* with the entry corresponding to an object attribute pair being assigned the value of that attribute which

3

the object satisfies. From the following definition, we can form equivalence relations on the objects for each given attribute. The set of these equivalence relations can then be used as our knowledge base.

Definition 2.1. *Let V be the set of objects and P be the set of attributes. Let $Q \subseteq P$, then V/Q is an equivalence relation on U induced by Q as follows: $x \sim_Q y$ iff $q(x) = q(y)$ for every $q \in Q$.*

To construct decision rules, we may fix two sets of attributes called *condition attributes* and *decision attributes* denoted by C and D respectively. We then use these to make predictions about the decision attributes based on the condition attributes. *Decision rules* are made by recording which values of decision attributes correlate with which values of condition attributes. As this information can be of considerable size, one of the primary goals of rough set theory is to reduce the number of decision attributes without losing predictive power. A minimal set of attributes which contains the same predictive power as the full set of decision attributes is called a *reduct* with respect to D.

Next we give the definition of the positive region of one equivalence relation with respect to another.

Definition 2.2. *Let C and D be equivalence relations on a finite non-empty set V. The* positive region *of the partition D with respect to C is given by,*

$$POS_C(D) = \bigcup_{X \in D} l_C(X),\tag{3}$$

Definition 2.3. *It is said that D depends on C in a degree k, where $0 \leq k \leq 1$, denoted by $C \Rightarrow_k D$, if*

$$k = \gamma(C, D) = \frac{|POS_C(D)|}{|V|}.\tag{4}$$

If $k = 1$, then we say that C depends totally on D i.e $C \Rightarrow D$.

Let $K_1 = (V, \mathcal{P})$ and $K_2 = (V, \mathcal{Q})$. We now give the definitions dependency of knowledge and then partial dependency. We say that $\mathcal{Q}$ *depends on* $\mathcal{P}$ i.e. $\mathcal{P} \Rightarrow \mathcal{Q}$ iff $IND(\mathcal{P}) \subseteq IND(\mathcal{Q})$.

Proposition 2.1. *$I_{IND(\mathcal{P})} \leq I_{IND(\mathcal{Q})}$ iff $\mathcal{P} \Rightarrow \mathcal{Q}$.*

Proposition 2.2. *$POS_{IND(\mathcal{P})}IND((\mathcal{Q})) = U$ iff $\mathcal{P} \Rightarrow \mathcal{Q}$.*

Otherwise, in the above case, $\gamma(IND(\mathcal{P}), IND(\mathcal{Q})) = k < 1$ and then we say that $\mathcal{P} \Rightarrow_k \mathcal{Q}$.

3   Properties of Successive Approximations

Next, we see that in general, approximating with respect to E_1 and then approximating the result with respect to E_2 gives a different result than if we had done it in the reverse order. That is, successive approximations do not commute. We consider some properties of successive approximations below.

Proposition 3.1. *Let V be a set and E_1 and E_2 be equivalence relations on V. Then for $Y \in \mathcal{P}(V)$, the following holds,*

1. $\boldsymbol{l}_{E_1}(\boldsymbol{l}_{E_2}(Y)) = Z \quad \not\Rightarrow \quad \boldsymbol{l}_{E_2}(\boldsymbol{l}_{E_1}(Y)) = Z,$
2. $\boldsymbol{u}_{E_1}(\boldsymbol{u}_{E_2}(Y)) = Z \quad \not\Rightarrow \quad \boldsymbol{u}_{E_2}(\boldsymbol{u}_{E_1}(Y)) = Z,$
3. $\boldsymbol{u}_{E_1}(\boldsymbol{l}_{E_2}(Y)) = Z \quad \not\Rightarrow \quad \boldsymbol{l}_{E_2}(\boldsymbol{u}_{E_1}(Y)) = Z,$
4. $\boldsymbol{l}_{E_1}(\boldsymbol{u}_{E_2}(Y)) = Z \quad \not\Rightarrow \quad \boldsymbol{u}_{E_2}(\boldsymbol{l}_{E_1}(Y)) = Z.$

Proof. We give a counterexample to illustrate the proposition. Let $V = \{a, b, c, d\}$ and let $E_1 = \{\{a, b, c\}, \{d\}\}$ and $E_2 = \{\{a, b\}, \{c, d\}\}$.
To illustrate 1., let $Y = \{a, b, c\}$. Then $\boldsymbol{l}_{E_1}(\boldsymbol{l}_{E_2}(Y)) = \emptyset$ while $\boldsymbol{l}_{E_2}(\boldsymbol{l}_{E_1}(Y)) = \{a, b\}$.

For 2., let $Y = \{a\}$. Then $\boldsymbol{u}_{E_1}(\boldsymbol{u}_{E_2}(Y)) = \{a, b, c\}$ while $\boldsymbol{u}_{E_2}(\boldsymbol{u}_{E_1}(Y)) = \{a, b, c, d\}$.

For 3., let $Y = \{a, b\}$. Then $\boldsymbol{u}_{E_1}(\boldsymbol{l}_{E_2}(Y)) = \{a, b, c\}$ while $\boldsymbol{l}_{E_2}(\boldsymbol{u}_{E_1}(Y)) = \{a, b\}$.

For 4., let $Y = \{a, b, c\}$. Then $\boldsymbol{l}_{E_1}(\boldsymbol{u}_{E_2}(Y)) = \emptyset$ while $\boldsymbol{u}_{E_2}(\boldsymbol{l}_{E_1}(Y)) = \{a, b, c, d\}$. $\qquad\square$

From Properties 1), 5) and 6) of lower and upper approximations in Section 2.1, we immediately get that,
(i) $\boldsymbol{l}_{E_1}(\boldsymbol{l}_{E_2}(Y)) \subseteq \boldsymbol{l}_{E_2}(Y)$, $\boldsymbol{u}_{E_1}(\boldsymbol{u}_{E_2}(Y)) \supseteq \boldsymbol{u}_{E_2}(Y)$,
   $\boldsymbol{u}_{E_1}(\boldsymbol{l}_{E_2}(Y)) \supseteq \boldsymbol{l}_{E_2}(Y)$ and $\boldsymbol{l}_{E_1}(\boldsymbol{u}_{E_2}(Y)) \subseteq \boldsymbol{u}_{E_2}(Y)$.

If we do not know anything more about the relationship between E_1 and E_2 then nothing further may be deduced. However, if for example we know that $E_1 \le E_2$ then the successive approximations are constrained as follows:

Proposition 3.2. *If $E_1 \le E_2$ then the following properties hold;*

(ii) $\boldsymbol{l}_{E_1}(\boldsymbol{l}_{E_2}(Y)) = \boldsymbol{l}_{E_2}(Y)$
(iii) $\boldsymbol{l}_{E_2}(\boldsymbol{l}_{E_1}(Y)) \subseteq \boldsymbol{l}_{E_2}(Y)$
(iv) $\boldsymbol{u}_{E_1}(\boldsymbol{u}_{E_2}(Y)) \supseteq \boldsymbol{u}_{E_1}(Y)$
(v) $\boldsymbol{u}_{E_2}(\boldsymbol{u}_{E_1}(Y)) = \boldsymbol{u}_{E_2}(Y)$

Proof. Straightforward. $\qquad\square$

Proposition 3.3. *Let V be a finite non-empty set and let E_1 and E_2 be equivalence relations on V. Let $x \in V$. Then $\boldsymbol{l}_{E_1}(\boldsymbol{u}_{E_2}(\{x\})) \subseteq POS_{E_1}(E_2)$.*

Corollary 3.1. *Let V be a finite non-empty set and let E_1 and E_2 be equivalence relations on V. Let $X \subseteq V$. Then $POS_{E_1}(E_2) \cap X \subseteq \bigcup_{x \in X} \boldsymbol{l}_{E_1}(\boldsymbol{u}_{E_2}(\{x\}))$.*

Corollary 3.2. *Let V be a finite non-empty set and let E_1 and E_2 be equivalence relations on V. Then $POS_{E_1}(E_2) = \bigcup_{x \in V} \boldsymbol{l}_{E_1}(\boldsymbol{u}_{E_2}(\{x\}))$.*
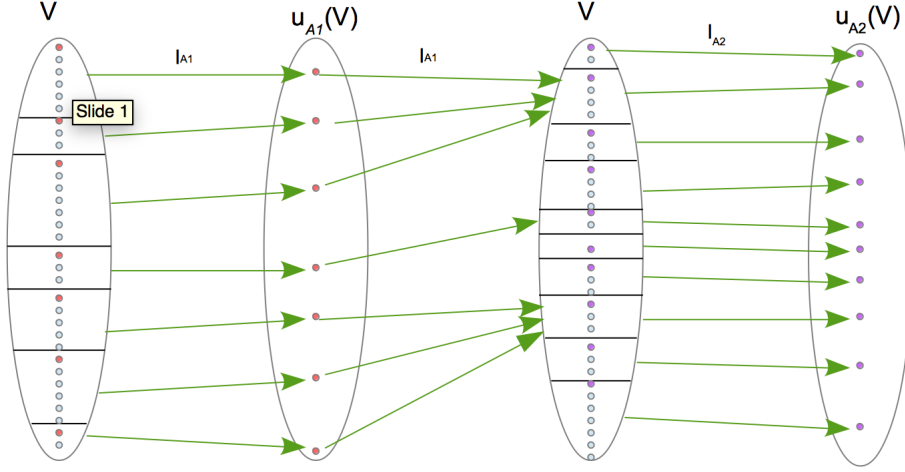
Figure 4.1: Illustrates that successive approximations get more coarse when iterated.

Proposition 3.4. *Let G be a graph with vertex set V and E an equivalence relation on V. Let S_E be the set containing equivalence classes of E and taking the closure under union. Let $F : \mathcal{P}(V) \to S_E$ be such that $F(X) = \bigcup_{x \in X} [x]_E$ and let $Id : S_E \to \mathcal{P}(V)$ be such that $Id(Y) = Y$ Then F and Id form a Galois connection.*

Proof. It is clear from definitions that both F and Id are monotone. We need that for $X \in \mathcal{P}(V)$ and $Y \in S_E$, $F(X) \subseteq Y$ iff $X \subseteq Id(Y)$. This is also the case because from the definition of F, we have the $X \subseteq F(X)$. $\qquad\square$

Remark 3.1. Successive approximations break the Galois structure of single approximations. We can imagine that single approximations are a kind of sorting on the domain of a structure. We partition objects in the domain into boxes and in each box there is a special member (the lower or upper approximation) which identifies/represents any member in its respective box. We may say that objects are approximated by their representative.

For successive approximations, we have two different sortings of the same domain. Objects are sorted by the first approximation and only their representative members are then sorted by the second approximation. An object is then placed in the box that its representative member is assigned to in the second approximation, even though the object itself may be placed differently if the second approximation alone was used. Hence the errors 'add' in some sense. In Figure 4.1, the final grouping as seen by following successive arrows, may be coarser than both the first and second approximations used singly. An interesting problem is how to correct/minimise these errors. It is also interesting how much of the individual approximations can be reconstructed from knowledge of the combined approximation. In the next section we will investigate this problem.

4   Decomposing $L_2 L_1$ Approximations

What if we knew that a system contained exactly two successive approximations? Would we be able to decompose them into its individual components? Before getting into what we can do and

what information can be extracted, we start with an example to illustrate this.

Notation: Let V be a finite set. Let a function representing the output of a subset of V when acted on by a lower approximation operator L_1 followed by a lower approximation operator L_2, based on the equivalence relations E_1 and E_2 respectively, be denoted by L_2L_1 where $L_2L_1(X) = L_2(L_1(X))$ and $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$. Similarly, other combinations of successive lower and upper approximations examined will be denoted by $U_2U_1,\ L_2U_1, U_2L_1$ which denotes successive upper approximations, an upper approximation followed by a lower approximation and a lower approximation followed by an upper approximation respectively.

Sometimes when we know that the approximations are based on equivalence relations P and Q we may use the subscripts to indicate this for example; L_QL_P.

Lastly, if for a defined L_2L_1 operator there exists a pair of equivalence relation solutions E_1 and E_2 which are such that the lower approximation operators L_1 and L_2 are based on them respectively, then we may denote this solution by the pair (E_1, E_2). Also, (E_1, E_2) can be said to produce or generate the operators based on them.

Example 4.1

Let $V = \{a, b, c, d, e\}$. Let a function representing the output of a subset of V when acted on by a lower approximation operator L_1 followed by a lower approximation operator L_2, which are induced by equivalence relations E_1 and E_2 respectively and let $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be as follows:

$L_2L_1(\{\emptyset\}) = \emptyset$ $L_2L_1(\{a, b, c, d, e\}) = \{a, b, c, d, e\}$

$L_2L_1(\{a\}) = \emptyset$ $L_2L_1(\{b, c, d, e\}) = \{e\}$

$L_2L_1(\{b\}) = \emptyset$ $L_2L_1(\{a, c, d, e\}) = \{c, d, e\}$

$L_2L_1(\{c\}) = \emptyset$ $L_2L_1(\{a, b, d, e\}) = \{e\}$

$L_2L_1(\{d\}) = \emptyset$ $L_2L_1(\{a, b, c, e\}) = \{a, b\}$

$L_2L_1(\{e\}) = \emptyset$ $L_2L_1(\{a, b, c, d\}) = \{a, b\}$

$L_2L_1(\{a, b\}) = \emptyset$ $L_2L_1(\{c, d, e\}) = \{e\}$

$L_2L_1(\{a, c\}) = \emptyset$ $L_2L_1(\{b, d, e\}) = \{e\}$

$L_2L_1(\{a, d\}) = \emptyset$ $L_2L_1(\{b, c, e\}) = \emptyset$

$L_2L_1(\{a, e\}) = \emptyset$ $L_2L_1(\{b, c, d\}) = \emptyset$

$L_2L_1(\{b, c\}) = \emptyset$ $L_2L_1(\{b, c, d\}) = \emptyset$

$L_2L_1(\{b, d\}) = \emptyset$ $L_2L_1(\{a, d, e\}) = \{e\}$

$L_2L_1(\{b, e\}) = \emptyset$ $L_2L_1(\{a, c, d\}) = \emptyset$

$L_2L_1(\{c, d\}) = \emptyset$ $L_2L_1(\{a, b, e\}) = \emptyset$

$L_2L_1(\{c, e\}) = \emptyset$ $L_2L_1(\{a, b, d\}) = \emptyset$

$L_2L_1(\{d, e\}) = \{e\}$ $L_2L_1(\{a, b, c\}) = \{a, b\}$

We will now try to reconstruct E_1 and E_2. The minimal sets in the output are $\{e\}$ and $\{a, b\}$. Clearly, these are either equivalence classes of E_2 or a union of two or more equivalence classes of E_2. Since $\{e\}$ is a singleton it is must be an equivalence class of E_2. So far we have partially reconstructed E_2 and it is equal to or finer than $\{\{a, b\}, \{c, d\}, \{e\}\}$.

Let us consider the pre-images of these sets in L_2L_1 to try to reconstruct E_1. Now, $L_2L_1^{-1}(\{e\}) = \{\{d, e\}, \{a, d, e\}, \{b, d, e\}, \{c, d, e\}, \{a, b, d, e\}, \{b, c, d, e\}\}$. We see that this set has a minimum with respect to containment and it is $\{d, e\}$. Hence either $\{d, e\}$ is an equivalence class of E_1 or both of $\{d\}$ and $\{e\}$ are equivalent classes of E_1.

Similarly, $L_2L_1^{-1}(\{a,b\}) = \{\{a,b,c\}, \{a,b,c,e\}, \{a,b,c,d\}\}$. We, see that this set has a minimum which is $\{a,b,c\}$ hence either this set is an equivalence class or is a union of equivalence classes in E_1. Now, $L_2L_1^{-1}(\{c,d,e\}) = \{\{a,c,d,e\}\}$. Hence, $\{a,c,d,e\}$ also consists of a union of equivalence classes of E_1. Since we know from above that $\{d,e\}$ consists of the union of one or more equivalence classes of E_1, this means that $\{a,c\}$ consists of the union of one or more equivalence classes of E_1 and $\{b\}$ is an equivalence class of E_1. So far we have that E_1 is equal to or finer than $\{\{a,c\}, \{b\}, \{d,e\}\}$.

Now we consider if $\{a,c\} \in E_1$ or both of $\{a\}$ and $\{c\}$ are in E_1. We can rule out the latter for suppose it was the case. Then $L_2L_1(\{a,b\})$ would be equal to $\{a,b\}$ since we already have that $\{b\} \in E_1$ and $\{a,b\}$ is the union of equivalence classes in E_2. Since this is not the case we get that $\{a,c\} \in E_1$. By a similar analysis of $L_2L_1(\{a,c,d\}) \neq \{c,d\}$ but only $\emptyset$ we get that $\{d,e\} \in E_1$. Hence, we have fully constructed E_1 and $E_1 = \{\{a,c\}, \{b\}, \{d,e\}\}$.

With E_1 constructed we can complete the construction of E_2. Recall, that we have that $\{a,b\}$ is a union of equivalence classes in E_2. Suppose that $\{a\} \in E_2$. Then $L_2L_1(\{a,c\})$ would be equal to $\{a\}$ since $\{a,c\} \in E_1$ but from the given list we see that it is not. Hence, $\{a,b\} \in E_2$. Similarly, we recall that $\{c,d\}$ is a union of equivalence classes in E_2. Suppose that $\{d\} \in E_2$. Then $L_2L_1(\{d,e\})$ would be equal to $\{d,e\}$ since $\{d,e\} \in E_1$ but it is only equal to $\{e\}$. Hence, $\{c,d\} \in E_2$. We have now fully reconstructed E_2 and $E_2 = \{\{a,b\}, \{c,d\}, \{e\}\}$.

The next example shows that we cannot always uniquely decompose successive approximations.

Example 4.2
Let $V = \{a,b,c,d\}$ and let $E_1 = \{\{a,b\}, \{c,d\}\}$, $E_2 = \{\{a,c\}, \{b,d\}\}$ and $E_3 = \{\{a,d\}, \{b,c\}\}$. We see that $L_1L_2(X) = L_1L_3(X) = \emptyset$ for all $X \in (\mathcal{P}(V) - V)$ and $L_1L_2(X) = L_1L_3(X) = V$ when $X = V$. Then for all $X \subseteq U$, $L_1L_2(X) = L_1L_3(X)$ even though $E_2 \neq E_3$. Hence, if we are given a double, lower successive approximation on $\mathcal{P}(V)$ which outputs $\emptyset$ for all $X \in (\mathcal{P}(V) - V)$ and V for $X = V$ then we would be unable to say that it was uniquely produced by L_1L_2 or L_1L_3.

In the following we start to build to picture of what conditions are needed for the existence of unique solutions for double, successive approximations.

Proposition 4.1. *Let V be a set with equivalence relations E_1 and E_2 on V. If for each $[x]_{E_1} \in E_1$, $[x]_{E_1}$ is such that $L_2([x]_{E_1}) = \emptyset$ i.e $[x]_{E_1}$ is either internally E_2–undefinable or totally E_2–undefinable, then the corresponding approximation operator, L_2L_1 on $\mathcal{P}(V)$ will be such that $L_2L_1([x]_{E_1}) = \emptyset$.*

Proof. Here, $L_1([x]) = \emptyset$. Hence $L_2L_1([x]) = L_2(\emptyset) = \emptyset$. $\square$

Remark 4.1 We note that the union of E-undefinable sets is not necessarily E-undefinable. Consider Example 4.2. Here, $\{a,b\}$ and $\{c,d\}$ are both totally E_2–undefinable but their union, $\{a,b,c,d\}$ is E_2–definable.

Algorithm 4.1: For Partial Decomposition of Double Successive Lower Approximations

Let V be a finite set. Given an input of a fully defined operator $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$, if a solution exists, we can produce a solution (S, R), i.e. where L_1 and L_2 are the lower approximation operators of equivalence relations S and R respectively, by performing the following steps:

1. Let J be the set of output sets of the given L_2L_1 operator. We form the relation R to be such that for $a, b \in V$, $a \sim_R b \iff (a \in X \iff b \in X)$ for any $X \in J$. It is clear that R is an equivalence relation.

2. For each $Y \neq \emptyset$ output set, find the minimum pre-image set with respect to $\subseteq$, Y_m, such that $L_2L_1(Y_m) = Y$. Collect all these minimum sets in a set K. If there is any non-empty output set Y, such that the minimum Y_m does not exist, then there is no solution to the given operator and we return 0 signifying that no solution exists.

3. Using K, we form the relation S to be such that for $a, b \in V$, $a \sim_S b \iff (a \in X \iff b \in X)$ for any $X \in K$. It is clear that S is an equivalence relation.

4. Form the operator $L_RL_S : \mathcal{P}(V) \to \mathcal{P}(V)$ generated by (S, R). If for all $X \in \mathcal{P}(V)$, the given L_2L_1 operator is such that $L_2L_1(X) = L_RL_S(X)$, then (S, R) is a solution proving that a solution exists (note that it is not necessarily unique). Return (S, R). Otherwise, discard S and R and return 0 signifying that no solution exists.

We will prove the claims in step 2 and step 4 in this section. Next, we prove step 2.

Proposition 4.2. *Let V be a set and $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given fully defined operator on $\mathcal{P}(V)$. If for $Y \neq \emptyset$ in the range of L_2L_1, there does not exist a minimum set Y_m, with respect to $\subseteq$ such that $L_2L_1(Y_m) = Y$, then there is no equivalence relation pair solution to the given operator.*

Proof. Suppose to get a contradiction that a solution (E_1, E_2) exists and there is no minimum set Y_m such that $L_2L_1(Y_m) = Y$. Since V is finite, then there exists at least two minimal sets Y_k and Y_l say, such that $L_2L_1(Y_s) = Y$ and $L_2L_1(Y_t) = Y$. Since Y_s and Y_t are minimal sets with the same output after two successive lower approximations, then Y_s and Y_t must each be unions of equivalence classes in E_1 which contain Y. Since they are unequal, then WLOG there exists $[a]_{E_1} \in E_1$ which is such that $[a]_{E_1} \in Y_s$ but $[a]_{E_1} \notin Y_t$. Since Y_s is minimal, then $[a]_{E_1} \cap Y \neq \emptyset$ (or else $L_2L_1(Y_s) = L_2L_1(Y_s - [a]_{E_1}) = Y$). So let $x \in [a]_{E_1} \cap Y$. Then $Y_t \not\supseteq x$ which contradicts $Y_t \supseteq Y$. $\square$

We now prove three lemmas on the way to proving the claim in step 4.

Lemma 4.1. *Let V be a set and $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given fully defined operator on $\mathcal{P}(V)$. Let R and S be equivalence relations defined on V as constructed in the previous algorithm. If (E_1, E_2) is a solution of L_2L_1 then $E_2 \leq R$ and $E_1 \leq S$.*

Proof. We first prove $E_2 \leq R$. Now the output set of a non-empty set in $\mathcal{P}(V)$ is obtained by first applying the lower approximation L_1 to it and and after applying the lower approximation, L_2 to it. Hence by definition of L_2, the non-empty output sets are unions of equivalence classes of the equivalence relation which corresponds to L_2. If a is in an output set but b is not then they cannot belong to the same equivalence class of E_2 i.e. $a \not\sim_R b$ implies that $a \not\sim_{E_2} b$. Hence $E_2 \leq R$.

Similarly, the minimal pre-image, X say, of a non-empty output set which is a union of equivalence classes in E_2, has to be a union of equivalence classes in E_1. For suppose it was not. Let $Y = \{y \in X \mid [y]_{E_1} \not\subseteq X\}$. By assumption, $Y \neq \emptyset$. Then $L_1(X) = L_1(X - Y)$. Hence $L_2L_1(X) = L_2L_1(X - Y)$ but $|X - Y| < |X|$ contradicting minimality of X. Therefore, if a belongs to the minimal pre-image of a non-empty output set but b does not belong to it, then a and b cannot belong to the same equivalence class in E_1 i.e. $a \not\sim_S b$ which implies that $a \not\sim_{E_1} b$. Hence $E_1 \leq S$. $\square$

9

Remark 4.2 The above lemma implies that for a given L_2L_1 operator on $\mathcal{P}(V)$ for a set V, that the pair of solutions given by the algorithm S and R for corresponding to L_1 and L_2 are the coarsest solutions for E_1 and E_2 which are compatible with the given, fully defined L_2L_1 operator. That is, for any other possible solutions, E_1 and E_2 to the given L_2L_1 operator, $E_1 \leq S$ and $E_2 \leq R$.

Lemma 4.2. *Let V be a finite set and $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a fully defined operator. If there exists equivalence pair solutions to the operator (E_1, E_2) which is such that there exists $[x]_{E_2}, [y]_{E_2} \in E_2$, such that $[x]_{E_2} \neq [y]_{E_2}$ and $\boldsymbol{u}_{E_1}([x]_{E_2}) = \boldsymbol{u}_{E_1}([y]_{E_2})$, then there exists another solution, (E_1, H_2), where H_2 is an equivalence relation formed from E_2 by combining $[x]_{E_2}$ and $[y]_{E_2}$ and all other elements are as in E_2. That is, $[x]_{E_2} \cup [y]_{E_2} = [z] \in H_2$ and if $[w] \in E_2$ such that $[w] \neq [x]_{E_2}$ and $[w]_{E_2} \neq [y]_{E_2}$, then $[w] \in H_2$.*

Proof. Suppose that (E_1, E_2) is a solution of a given L_2L_1 operator and H_2 is as defined above. Now, $L_2L_1(X) = Y$ iff the union of E_1-equivalence classes in X contains the union of E_2-equivalence classes which is equal to Y. So, in the (E_1, H_2) solution, the only way that $L_{H_2}L_{E_1}(X)$ could be different from $L_{E_2}L_{E_1}(X)$(which is $= L_2L_1(X)$) is if (i) $[x]_{E_2}$ is contained in $L_{E_2}L_{E_1}(X)$ while $[y]_{E_2}$ is not contained in $L_{E_2}L_{E_1}(X)$ or if (ii) $[y]_{E_2}$ is contained in $L_{E_2}L_{E_1}(X)$ while $[x]_{E_2}$ is not contained in $L_{E_2}L_{E_1}(X)$. This is because in H_2, $[x]_{E_2}$ and $[y]_{E_2}$ always occur together in an output set if they are in it at all (recall that output sets are unions of equivalence classes) in the equivalence class of $[z] = [x]_{E_2} \cup [y]_{E_2}$ and all the other equivalence classes of H_2 are the same as in E_2. However, neither (i) nor (ii) is the case since $\boldsymbol{u}_{E_1}([x]_{E_2}) = \boldsymbol{u}_{E_1}([y]_{E_2})$. That is, the equivalence classes of $[x]_{E_2}$ are contained by exactly the same union of equivalences in E_1 which contains $[y]_{E_2}$. Thus, any set X which contains a union of E_1-equivalences which contains $[x]_{E_2}$ also must contain $[y]_{E_2}$ and therefore $[z]_H$. Hence, if (E_1, E_2) is a solution for the given vector, then so is (E_1, H_2). $\square$

Lemma 4.3. *Let V be a finite set and $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a fully defined operator. If there exists equivalence pair solutions to the operator (E_1, E_2) which is such that there exists $[x]_{E_1}, [y]_{E_1} \in E_2$, such that $[x]_{E_1} \neq [y]_{E_1}$ and $\boldsymbol{u}_{E_2}([x]_{E_1}) = \boldsymbol{u}_{E_2}([y]_{E_1})$, then there exists another solution, (H_1, E_2), where H_1 is an equivalence relation formed from E_1 by combining $[x]_{E_2}$ and $[y]_{E_2}$ and all other elements are as in E_1. That is, $[x]_{E_1} \cup [y]_{E_1} = [z] \in H_1$ and if $[w] \in E_2$ such that $[w] \neq [x]_{E_1}$ and $[w]_{E_1} \neq [y]_{E_1}$, then $[w] \in H_1$.*

Proof. Suppose that (E_1, E_2) is a solution of a given L_2L_1 operator and H_1 is as defined above. Now, $L_2L_1(X) = Y$ iff the union of E_1-equivalence classes in X contains the union of E_2-equivalence classes which is equal to Y. So, in the (H_1, E_2) solution, the only way that $L_{E_2}L_{H_1}(X)$ could be different from $L_{E_2}L_{E_1}(X)$(which is $= L_2L_1(X)$) is if the union of equivalence classes in X which is needed to contain Y, (i) contains $[x]_{E_2}$ but not $[y]_{E_2}$ or (ii) contains $[y]_{E_2}$ but not $[z]_{E_2}$. However, this is not the case since $\boldsymbol{u}_{E_2}([x]_{E_1}) = \boldsymbol{u}_{E_2}([y]_{E_1})$. That is, $[x]_{E_1}$ intersects exactly the same equivalence classes in E_2 as $[y]_{E_1}$. So if $[x]_{E_1}$ is needed to contain an equivalence class in E_2, then $[y]_{E_1}$ is also needed. In other words, if $L_2L_1(X) = Y$, then for any minimal set such $Y_m \subseteq X$ such that $L_2L_1(Y_m) = Y$, $[x]_{E_1}$ is contained in Y_m iff $[y]_{E_1}$ is contained in Y_m iff $[z] \in H_1$ is contained in Y_m. Hence, if (E_1, E_2) is a solution for the given vector, then so is (H_1, E_2). $\square$

We now have enough to be able to prove the claim in step 4 of Algorithm 4.1 (actually we prove something stronger because we also show conditions which the solutions of the algorithm must satisfy).

Theorem 4.1. *Let V be a finite set and $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a fully defined operator. If there exists equivalence pair solutions to the operator, then there exists solutions (E_1, E_2) which satisfy,*

(i) for each $[x]_{E_2}, [y]_{E_2} \in E_2$, if $[x]_{E_2} \neq [y]_{E_2}$ then $\boldsymbol{u}_{E_1}([x]_{E_2}) \neq \boldsymbol{u}_{E_1}([y]_{E_2})$,

(ii) for each $[x]_{E_1}, [y]_{E_1} \in E_1$, if $[x]_{E_1} \neq [y]_{E_1}$ then $\boldsymbol{u}_{E_2}([x]_{E_1}) \neq \boldsymbol{u}_{E_2}([y]_{E_1})$.

Furthermore, $E_1 = S$ and $E_2 = R$ where (S, R) are the solutions obtained by applying Algorithm 4.1 to the given L_2L_1 operator.

Proof. Suppose that there exists a solution (C, D). Then, either (C, D) already satisfies condition (i) and condition (ii) or it does not. If it does, take $(E_1, E_2) = (C, D)$. If it does not satisfy condition (i) then use repeated applications of Lemma 4.2 until we arrive at an (C, E_2) solution which does. Similarly, if (C, E_2) does not satisfy condition (ii), use repeated applications of Lemma 4.3 until it does. Since $\mathcal{P}(V)$ is finite this will take at most finite applications of the lemmas until we obtain a solution, (E_1, E_2) which satisfies the conditions of the theorem. Since there is a solution, using Proposition 4.2 we will at least be able to reach step 4 of Algorithm 4.1. So let S and R be the relations formed by the algorithm after step 3. Next, we will show that $E_1 = S$ and $E_2 = R$. Now, by Lemma 4.1, we have that $E_1 \leq S$ and $E_2 \leq R$.

Consider the output sets of the given L_2L_1 operator. It is clear that these sets are unions of one or more equivalence classes of E_2. Let $[y]_{E_2} \in E_2$ then $L_2L_1(\boldsymbol{u}_{E_1}([y]_{E_2})) \supseteq [y]_{E_2}$.

Claim 1: $L_2L_1(\boldsymbol{u}_{E_1}([y]_{E_2}))$ is the minimum output set of L_2L_1 such that it contains $[y]_{E_2}$ and $\boldsymbol{u}_{E_1}([y]_{E_2})$ is the minimum set X such that $L_2L_1(X) \supseteq [y]_{E_2}$.

To see this we first note that L_2L_1 is a monotone function on $\mathcal{P}(V)$ since L_1 and L_2 are monotone operators and L_2L_1 is the composition of them. Then, if we can show that $\boldsymbol{u}_{E_1}([y]_{E_2})$ is the minimum set $X \in \mathcal{P}(V)$, such that $L_2L_1(X) \supseteq [y]_{E_2}$, then $L_2L_1(\boldsymbol{u}_{E_1}([y]_{E_2}))$ will be the minimum set output set which contains $[y]_{E_2}$. This is true because for $L_2L_1(X) \supseteq [y]_{E_2}$, then $L_1(X)$ must contain each member of $[y]_{E_2}$. We note that the range of L_1 contains only unions of equivalence classes of E_1 (counting the emptyset as a union of zero sets). Hence for $L_1(X)$ to contain each element of $[y]_{E_2}$, it must contain each equivalence class in E_1 which contains any of these elements. In other words, it must contain $\boldsymbol{u}_{E_1}([y]_{E_2})$. Suppose that X is such that $X \not\supseteq \boldsymbol{u}_{E_1}([y]_{E_2})$ and $L_2L_1(X) \supseteq [y]_{E_2}$. Then for some $v \in [y]_{E_2}$, v is not in X and so $\boldsymbol{u}_{E_1}([v]_{E_2}) \notin L_1(X)$. Hence $L_2L_1(X) \not\supseteq v$ and so does not contain $[y]_{E_2}$ which is a contradiction.

Claim 2: $L_2L_1(\boldsymbol{u}_{E_1}([y]_{E_2}))$ is not the minimum output set with respect to containing any other $[z]_{E_2} \neq [y]_{E_2}$.

Suppose that for some $[z]_{E_2} \neq [y]_{E_2} \in E_2$, that $L_2L_1(\boldsymbol{u}_{E_1}([y]_{E_2}))$ is the minimum output set containing $[z]_{E_2}$. Then by the previous Claim, we get that $L_2L_1(\boldsymbol{u}_{E_1}([y]_{E_2})) = L_2L_1(\boldsymbol{u}_{E_1}([z]_{E_2}))$ and that $\boldsymbol{u}_{E_1}([y]_{E_2}) \supseteq \boldsymbol{u}_{E_1}([z]_{E_2})$. But since $\boldsymbol{u}_{E_1}([y]_{E_2})$ is the minimum set such that $L_2L_1(X) \supseteq [y]_{E_2}$, then the stated equality also gives us that $\boldsymbol{u}_{E_1}([y]_{E_2}) \subseteq \boldsymbol{u}_{E_1}([z]_{E_2})$. Hence we have $\boldsymbol{u}_{E_1}([y]_{E_2}) = \boldsymbol{u}_{E_1}([z]_{E_2})$ which is a contradiction to the assumption of condition (i) of the theorem.

Now we can reconstruct E_2 by relating elements which always occur together in the output sets. That is, $a \sim_R b \iff (a \in X \iff b \in X)$ for each X in the range of L_2L_1. From the previous proposition we have that $E_2 \leq R$. We claim that $R \leq E_2$, hence $R = E_2$. To show this, suppose that it is not the case. Then there exists $a, b \in V$ such that $a \sim_R b$ but $a \not\sim_{E_2} b$. By Claim 1, $L_2L_1(\boldsymbol{u}_{E_1}([a]_{E_2}))$ is the minimum set which contains $[a]_{E_2}$ and since $a \sim_R b$ then it must contain b, and consequently $[b]_{E_2}$ as well. Similarly by Claim 1, $L_2L_1(\boldsymbol{u}_{E_1}([b]_{E_2}))$ is the

minimum set which contains $[b]_{E_2}$ and since $a \sim_R b$ then it must contain a, and consequently $[a]_{E_2}$ as well. By minimality we therefore have both $L_2 L_1(\boldsymbol{u}_{E_1}([a]_{E_2})) \subseteq L_2 L_1(\boldsymbol{u}_{E_1}([b]_{E_2}))$ and $L_2 L_1(\boldsymbol{u}_{E_1}([a]_{E_2})) \supseteq L_2 L_1(\boldsymbol{u}_{E_1}([b]_{E_2}))$ which implies that $L_2 L_1(\boldsymbol{u}_{E_1}([a]_{E_2})) = L_2 L_1(\boldsymbol{u}_{E_1}([b]_{E_2}))$. This contradicts Claim 2 since $[a]_{E_2} \neq [b]_{E_2} \in E_2$. Hence, $E = R$ and we can reconstruct E_2 by forming the equivalence relation R which was defined by using the output sets.

It remains to reconstruct E_1. Next, we list the pre-images of the minimal output sets which contain $[y]_{E_2}$ for each $[y]_{E_2}$ in E_2 and by Claim 1 this exists and is equal to $\boldsymbol{u}_{E_1}([y]_{E_2})$. This implies that each such set is the union of some of the equivalence classes of E_1. Now using this pre-image list we relate elements of V in the following way: $a \sim_S b \iff (a \in X \iff b \in X)$ for each X in the pre-image list. From the previous proposition we have that $E_1 \leq S$. We claim that $S \leq E_1$ and hence $S = E_1$. Suppose that it was not the case. That is, there exists $a, b \in V$ such that $a \sim_S b$ but $a \nsim_{E_1} b$. Hence $[a]_{E_1} \neq [b]_{E_1}$. By condition (ii) of the theorem, we know that $\boldsymbol{u}_{E_2}([a]_{E_1}) \neq \boldsymbol{u}_{E_2}([b]_{E_1})$. So WLOG suppose that $d \in \boldsymbol{u}_{E_2}([a]_{E_1})$ but $d \notin \boldsymbol{u}_{E_2}([b]_{E_1})$. Since these sets are unions of equivalence classes in E_2 this implies that 1), $[d]_{E_2} \subseteq \boldsymbol{u}_{E_2}([a]_{E_1})$ and 2) $[d]_{E_2} \cap \boldsymbol{u}_{E_2}([b]_{E_1}) = \emptyset$. Now by Claim 1, $\boldsymbol{u}_{E_1}([d]_{E_2})$ is the minimum set, X such that $L_2 L_1(X)$ contains $[d]_{E_2}$ and so is on the output list from which the Relation S was formed. However, 1) implies that this set contains a while 2) implies that this set does not contain b. This contradicts $a \sim_S b$. Hence $S = E_1$ and we can construct E_1 by constructing S. The result is shown. $\square$

Next we give, a graph-theoretic equivalence of the theorem but we first define a graph showing the relationship between two equivalence relations on a set.

Definition 4.1. *Let C and D be two equivalence relations on a set V. Form a bipartite graph $B(C, D) = (G, E)$, where the nodes G is such that $G = \{[u]_C \mid [u]_C \in C\} \cup \{[u]_D \mid [u]_D \in D\}$ and the edges E are such that $E = \{([u]_C, [v]_D) \mid \exists x \in V : x \in [u]_C \text{ and } x \in [v]_D\}$. We call this the* **incidence graph** *of the pair (C, D).*

Theorem 4.2. *Let V be a finite set and let $L_2 L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given fully defined operator on $\mathcal{P}(V)$. If there exists solutions (E_1, E_2) then the incidence graph of E_1 and E_2, $B(E_1, E_2)$, is such that there are no compete bipartite subgraphs as components other than edges (or K_2).*

Proof. This is a direct translation of the previous theorem graph-theoretically. Suppose that the incidence graph of E_1 and E_2, $B(E_1, E_2)$, contains a complete bipartite subgraph as a component. Then the partition corresponding to E_2 violates Condition (i) of the theorem and the partition corresponding to E_1 violates condition (ii) of the theorem. $\square$

Corollary 4.1. *Let V be a finite set and $L_2 L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given defined operator. If (E_1, E_2) is a unique solution for the operator then $|E_1| < 2^{|E_2|}$ and $|E_2| < 2^{|E_1|}$.*

Proof. This follows directly from the conditions since in the incidence graph of a unique solution (E_1, E_2), each equivalence class in E_1 is mapped to a unique non-empty subset of equivalence classes in E_2 and vice versa. $\square$

The next natural question is, without assuming conditions on the equivalence relations, are there instances when the algorithm produces a unique solution? Example 4.1 is an example of a unique decomposition of a given $L_2 L_1$ operator. So this leads naturally to the next question. What conditions result in a unique solution to a given $L_2 L_1$? Can we find characterising features of the pairs of equivalences relations which give a unique $L_2 L_1$ operator?

12

We note that the algorithm always produces a solution for a fully defined L_2L_1 operator which has at least one solution. Hence, if there is a unique solution then these pairs of equivalence relations satisfy the conditions of Theorem 4.1. Recall that in Example 4.2, we were given an L_2L_1 operator defined on $\mathcal{P}(V)$ for $V = \{a, b, c, d\}$ such that $L_2L_1(X) = \emptyset$ for all $X \neq V$ and $L_2L_1(V) = V$. This example shows us that in addition to a solution which would satisfy the conditions of the theorem, which applying the algorithm gives us; $E_1 = \{\{a, b, c, d\}\}$ and $E_2 = \{\{a, b, c, d\}\}\}$, we also have solutions of the form $E_1 = \{\{a, b\}, \{c, d\}\}$ and $E_2 = \{\{a, c\}, \{b, d\}\}$ or $E_1 = \{\{a, b\}, \{c, d\}\}$ and $E_2 = \{\{a, d\}, \{b, c\}\}$ amongst others. In Lemma 4.1, we showed that the solution given by the algorithm is the coarsest pair compatible with a given defined L_2L_1 operator. We now try to find a condition such that after applying the algorithm, we may deduce whether or not the (S, R) solution is unique. This leads us to the next section.

4.1 Characterising Unique Solutions

Theorem 4.3. *Let V be a finite set and let $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a fully defined operator on $\mathcal{P}(V)$. If (S, R) is returned by Algorithm 4.1, then (S, R) is the unique solution of the operator iff the following holds:*

(i) For any $[x]_R \in R$, there exists $[z]_S \in S$ such that, $|[x]_R \cap [z]_S| = 1$.
(ii) For any $[x]_S \in S$, there exists $[z]_R \in R$ such that, $|[x]_S \cap [z]_R| = 1$.

Proof. We prove $\Leftarrow$ direction first. So assume the conditions. We note that by Lemma 4.1, any other solutions, (E_1, E_2) to the given L_2L_1 operator must be coarser than (S, R). Thus, if there is another solution to the given L_2L_1 operator, (E_1, E_2) then at least one of $E_1 < S$, $E_2 < R$ must hold.

First we assume to get a contradiction that there exists a solution (E_1, E_2) which is such that $E_1 < S$. That is, E_1 contains a splitting of at least one of the equivalences classes of S, say $[a]_S$. Hence $|[a]_S| \geq 2$. By assumption there exists a $[z]_R \in R$ such that $|[a]_S \cap [z]_R| = 1$. Hence there is a $[z]_{E_2} \in E_2$ such that $|[a]_S \cap [z]_{E_2}| = 1$ since $E_2 \leq R$. Call the element in this intersection v say. We note that $[v]_{E_2} = [z]_{E_2}$. Now as $[a]_S$ is spilt into smaller classes in E_1, v must be in one of these classes, $[v]_{E_1}$. Consider the minimal pre-image of the minimal output set of L_2L_1 which contains $[v]_R$. Call this set $Y_{(S,R)}$. For the solution (S, R), $Y_{(S,R)}$ contains all of $[a]_S$ since $v \in [a]_S$. But for the solution (E_1, E_2), the minimal pre-image of the minimal output set of L_2L_1 which contains $[v]_R$, $Y_{(E_1, E_2)}$, is such that $Y_{(E_1, E_2)} = (Y_S - [a]_s) \cup [v]_{E_1} \neq Y_S$. Hence the output list for (E_1, S) is different from the given one which is a contradiction.

Next, suppose to get a contradiction there exists a solution (E_1, E_2) which is such that $E_2 < R$. That is, E_2 contains a splitting of at least one of the equivalences classes of R, say $[a]_R$. Hence $|[a]_R| \geq 2$. By assumption there exists a $[z]_S \in S$ such that $|[a]_R \cap [z]_S| = 1$. Hence there is a $[z]_{E_1} \in E_1$ such that $|[a]_R \cap [z]_{E_1}| = 1$ since $E_1 \leq S$. Call the element in this intersection v say. We note that $[v]_{E_1} = [z]_{E_1}$. Now as $[a]_R$ is spilt into smaller classes in E_2, v must be in one of these classes, $[v]_{E_2}$. Consider the set $[a]_R - [v]_{E_2}$. The minimal pre-image of the minimal output set which contains this set in the (S, R) solution, $Y_{(S,R)}$ contains $[v]_S$ since here the minimal output set which contains $([a]_R - [v]_{E_2})$, must contain all of $[a]_R$ which contains v. If (E_1, E_2) were the solution then the minimal pre-image of the minimal output set which contains $([a]_R - [v]_{E_2})$, $Y_{(E_1, E_2)}$, would not contain $[v_s]$ since $([a]_R - [v]_{E_2}) \cap [v]_S = \emptyset$. That is, $Y_{(E_1, E_2)} \neq Y_S$. Hence the output list for (E_1, E_2) is different from the given one which is a contradiction.

13

Now we prove $\Rightarrow$ direction. Suppose that (E_1, E_2) is the unique solution, and assume that the condition does not hold. By Theorem 4.1, $(E_1, E_2) = (S, R)$. Then either there exists an $[x]_R \in R$ such that for all $[y]_S \in S$ such that $[x]_R \cap [y]_S \neq \emptyset$ we have that $|[x]_R \cap [y]_S| \geq 2$ or there exists an $[x]_S \in S$ such that for all $[y]_R \in R$ such that $[x]_S \cap [y]_R \neq \emptyset$ we have that $|[x]_S \cap [y]_R| \geq 2$.

We consider the first case. Suppose that $[x]_R$ has non-empty intersection with with n sets in S. We note that $n \geq 1$. Form a sequence of these sets; $S_1, ...S_n$. Since $|[x]_R \cap S_i| \geq 2$ for each i such that $i = 1, ...n$, let $\{a_{i1}, a_{i2}\}$ be in $[x]_R \cap S_i$ for each i such that $i = 1, ...n$. We split $[x]_R$ to form a finer E_2 as follows: Let $P = \{a_{i1} \mid i = 1, ...n\}$ and $Q = [x]_R - P$ be equivalence classes in E_2 and for the remaining equivalence classes in E_2, let $[y] \in E_2$ iff $[y] \in R$ and $[y]_R \neq [x]_R$. Now, $L_R L_S(X) = Y$ iff the union of S-equivalence classes in X contains the union of R-equivalence classes which is equal to Y. So, for the (S, E_2) solution, the only way that $L_{E_2} L_S(X)$ could be different from $L_R L_S(X)$ is if there is a union of S-equivalence classes in X which contain P but not Q or which contain Q but not P (since P and Q always occur together as $[x]_R$ for the (S, R) solution). However, this is not the case as follows. Since P and Q exactly spilt all of the equivalence classes of S which have non-empty intersection with $[x]_R$, we have that $\boldsymbol{u}_S(P) = \boldsymbol{u}_S(Q)$. That is, P intersects exactly the same equivalence classes of S as Q. Therefore, P is contained by exactly the same union of equivalence classes in S as Q. Therefore, a union of S-equivalence classes in X contains P iff it contains Q iff its contains $[x]_R$. Hence, $L_R L_S(X) = L_{E_2} L_S(X)$ for all $X \in \mathcal{P}(V)$ and if (S, R) is a solution for the given vector, then so is (S, E_2) which is a contradiction of assumed uniqueness of (S, R).

We consider the second case. Suppose that $[x]_S$ has non-empty intersection with with n sets in R. We note that $n \geq 1$. Form a sequence of these sets; $R_1, \ldots R_n$. Since $|[x]_S \cap R_i| \geq 2$ for each i such that $i = 1, \ldots n$, let $\{a_{i1}, a_{i2}\}$ be in $[x]_S \cap R_i$ for each i such that $i = 1, \ldots n$. We split $[x]_S$ to form a finer E_1 as follows: Let $P = \{a_{i1} \mid i = 1, \ldots n\}$ be one equivalence class and let $Q = [x]_R - P$ be another and for any $[y]_S \in S$ such that $[y]_S \neq [x]_S$, let $[y] \in E_1$ iff $[y] \in S$. Again, $L_R L_S(X) = Y$ iff the union of S-equivalence classes in X contains the union of R-equivalence classes which is equal to Y. So, for the (E_1, R) solution, the only way that $L_R L_{E_1}(X)$ could be different from $L_R L_S(X)$ is if (i) P is contained in $L_R L_S(X)$ while Q is not contained in $L_R L_S(X)$ or (ii) Q is contained in $L_R L_S(X)$ while P is not contained in $L_R L_S(X)$. Since P and Q spilt all of the equivalence classes of R which have non-empty intersection with $[x]_S$, this implies that $\boldsymbol{u}_R(P) = \boldsymbol{u}_R(Q)$. That is, P and Q intersect exactly the same equivalence classes of R. So if P is needed to contain an equivalence class in R for the (S, R) solution, then Q is also needed. In other words, if $L_2 L_1(X) = Y$, then for any minimal set such $Y_m \subseteq X$ such that $L_2 L_1(Y_m) = Y$, P is contained in Y_m iff Q is contained in Y_m iff $[x]_S$ is contained in Y_m. Hence, $L_R L_S(X) = L_R L_{E_1}(X)$ for all $X \in \mathcal{P}(V)$ and if (S, R) is a solution for the given vector, then so is (E_1, R) which is a contradiction of assumed uniqueness of (S, R). $\qquad \square$

The following theorem sums up the results of Theorem 4.1 and Theorem 4.3.

Theorem 4.4. *Let V be a finite set and let $L_2 L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a fully defined successive approximation operator on $\mathcal{P}(V)$. If (E_1, E_2) is a solution of the operator then it is the unique solution iff the following holds:*

(i) For each $[x]_{E_2}, [y]_{E_2} \in E_2$, if $[x]_{E_2} \neq [y]_{E_2}$ then $\boldsymbol{u}_{E_1}([x]_{E_2}) \neq \boldsymbol{u}_{E_1}([y]_{E_2})$,

(ii) For each $[x]_{E_1}, [y]_{E_1} \in E_1$, if $[x]_{E_1} \neq [y]_{E_1}$ then $\boldsymbol{u}_{E_2}([x]_{E_1}) \neq \boldsymbol{u}_{E_2}([y]_{E_1})$.

(iii) For any $[x]_{E_2} \in E_2$, there exists $[z]_{E_1} \in E_1$ such that, $|[x]_{E_2} \cap [z]_{E_1}| = 1$.

14

(iv) For any $[x]_{E_1} \in E_1$, there exists $[z]_{E_2} \in E_2$ such that, $|[x]_{E_1} \cap [z]_{E_2}| = 1$.

Remark 4.3: If an equivalence relation pair satisfies the conditions of Theorem 4.1, then the L_2L_1 operator based on those relations would be such that if there exists other solutions then they would be finer pairs of equivalence relations. On the other hand, if an equivalence relation pair satisfies the conditions of Theorem 4.3, then the L_2L_1 operator based on those relations would be such that if there exists other solutions then they would be coarser pairs of equivalence relations. Hence, if an equivalence relation pair satisfies the conditions of both Theorem 4.1 and Theorem 4.3, then the L_2L_1 operator produced by it is unique.

Corollary 4.2. *Let V be a finite set and let $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a fully defined successive approximation operator on $\mathcal{P}(V)$. If (S, R) is the solution returned by Algorithm 4.1, is such that it is the unique solution then following holds:*

For any $x \in V$ we have that;
(i) $[x]_S \not\supseteq [x]_R$ unless $|[x]_R| = 1$
(ii) $[x]_R \not\supseteq [x]_S$ unless $|[x]_S| = 1$,

Proof. This follows directly from the conditions in Theorem 4.3. ☐

Example 4.1 (*revisited*): Consider again, the given output vector of Example 4.1. First we form the (S, R) pair using Algorithm 4.1. We get that $R = \{\{a, b\}, \{c, d\}, \{e\}\}$ and $S = \{\{a, c\}, \{b\}, \{d, e\}\}$. Since this is the pair produced from Algorithm 4.1, we know that it satisfies the conditions of Theorem 4.1. Now we need only to check if this pair satisfies the conditions of Theorem 4.3 to see if it is the only solution to do so. To keep track of which equivalence class a set belongs to, we will index a set belonging to either S or R by S or R respectively. Then we see that $|\{a, b\}_R \cap \{b\}_S| = 1$, $|\{c, d\}_R \cap \{a, c\}_S| = 1$ and $|\{e\}_R \cap \{d, e\}_S| = 1$. This verifies both conditions of Theorem 4.3 and therefore this is the unique solution of the given operator.

Proposition 4.3. *Let V be a finite set and $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given defined operator. If (E_1, E_2) is a unique solution such that either $E_1 \neq Id$ or $E_2 \neq Id$ where Id is the identity equivalence relation on V then,*

(i) $E_1 \not\leq E_2$,
(ii) $E_2 \not\leq E_1$.

Proof. We first observe that if E_1 and E_2 are unique solutions and both of them are not Id then one of them cannot be equal Id. This is because if (E_1, Id) were solutions to a given L_2L_1 operator corresponding to L_1 and L_2 respectively then (Id, E_1) would also be solutions corresponding to L_1 and L_2 respectively and the solutions would not be unique. Hence, each of E_1 and E_2 contains at least one equivalence class of size greater than or equal to two.

Suppose that $E_1 \leq E_2$. Consider an $e \in E_2$ such that $|e| \geq 2$. Then e either contains a $f \in E_1$ such that $|f| \geq 2$ or two or more singletons in E_1. Then first violates the condition of Corollary 4.2 and the second violates the second condition of Theorem 4.1. Hence the solutions cannot be unique. Similarly, if we suppose that $E_2 \leq E_1$. ☐

Corollary 4.3. *Let V be a finite set and $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given defined operator. If there exists a unique solution (E_1, E_2) such that either $E_1 \neq Id$ or $E_2 \neq Id$ where Id is the identity equivalence relation on V then,*

(i) $k = \gamma(E_1, E_2) = \frac{|POS_{E_1}(E_2)|}{|V|} < 1$ *or* $E_1 \nRightarrow E_2$

(ii) $k = \gamma(E_2, E_1) = \frac{|POS_{E_2}(E_1)|}{|V|} < 1$ *or* $E_2 \nRightarrow E_1$.

Proof. This follows immediately from definitions. $\square$

Proposition 4.4. *Let V be a finite set and $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given defined operator. If there exists exists a unique solution (E_1, E_2) then,*

(i) for any $[x]_{E_1} \in E_1$, $|POS_{E_2}([x]_{E_1})| \leq 1$
(ii) for any $[x]_{E_2} \in E_2$, $|POS_{E_1}([x]_{E_2})| \leq 1$.

Proof. This follows from the conditions in Theorem 4.4 and Corollary 4.2 which imply that for a unique pair solution (E_1, E_2), an equivalence class of one of the equivalence relations cannot contain any elements of size greater than one of the other relation and can contain at most one element of size exactly one of the other relation. $\square$

Corollary 4.4. *Let V be a finite set where $|V| = l$ and $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given defined operator. If there exists exists a unique solution (E_1, E_2) such that $|E_1| = n$ and $|E_2| = m$ then,*

(i) $k = \gamma(E_1, E_2) = \frac{|POS_{E_1}(E_2)|}{|V|} \leq \frac{m}{l}$

(ii) $k = \gamma(E_2, E_1) = \frac{|POS_{E_2}(E_1)|}{|V|} \leq \frac{n}{l}$.

Proof. Let (E_1, E_2) be the unique solution of the given L_2L_1 operator. This result follows directly from the previous proposition by summing over all the elements in one member of this pair for taking its positive region with respect to the other member of the pair. $\square$

Corollary 4.5. *Let V be a finite set such that $|V| = n$ and $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given defined operator. If there exists exists a unique solution (E_1, E_2) then,*

(i) if the minimum size of an equivalence class in E_1, k_1 where $k_1 \geq 2$ then
$k = \gamma(E_1, E_2) = \frac{|POS_{E_1}(E_2)|}{|V|} = 0$.

(ii) if the minimum size of an equivalence class in E_2, k_2 where $k_2 \geq 2$ then
$k = \gamma(E_2, E_1) = \frac{|POS_{E_2}(E_1)|}{|V|} = 0$.

Proof. Since no member of E_2 can contain any member of E_1 because E_1 has no singletons, we get that $\frac{|POS_{E_1}(E_2)|}{|V|} = 0$. Similarly for Part (ii). $\square$

Proposition 4.5. *Let V be a finite set and $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given defined operator. If there exists a unique solution (E_1, E_2) such that $|E_1| = m$ and $|E_2| = n$ and S_1 is the number of singletons in E_1 and S_2 is the number of singletons in E_2, then,*

(i) $S_1 \leq n$
(ii) $S_2 \leq m$.

Proof. We note that the conditions in Theorem 4.4 imply that no two singletons in E_1 can be contained by any equivalence class in E_2 and vice versa. The result thus follows on application of the pigeonhole principle between the singletons in one equivalence relation and the number of elements in the other relation. $\square$

4.2 A Derived Preclusive Relation and a Notion of Independence

In [4], Cattaneo and Ciucci found that preclusive relations are quite useful for using rough approximations in information systems. In this direction, we will define a related notion of independence of equivalence relations from it.

Let V be a finite set and let $\mathfrak{E}_V$ be the set of all equivalence relations on V. Also, let $\mathfrak{E}_V^0 = \mathfrak{E}_V - Id_V$, where Id_V is the identity relation on V. From now on, where the context is clear, we will omit the subscript. We now define a relation on $\mathfrak{E}^\circ$, $\not\Rightarrow_{\mathfrak{E}^\circ}$, as follows:

Let E_1 and E_2 be in $\mathfrak{E}^\circ$. Let $L_2 L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ where L_1 and L_2 are lower approximation operators based on E_1 and E_2 respectively. Then,

$$E_1 \not\Rightarrow_{\mathfrak{E}^\circ} E_2 \text{ iff } L_2 L_1 \text{ is a unique approximation operator.}$$

That is, if for no other E_3 and E_4 in $\mathfrak{E}^\circ$ where at least one of $E_1 \neq E_3$ or $E_2 \neq E_4$ holds, is it the case that the operator $L_2 L_1 = L_3 L_4$, where L_3 and L_4 are lower approximation operators based on E_3 and E_4 respectively.

Definition 4.2. *Let V be a set and $E_1, E_2 \in \mathfrak{E}_V^0$. We say that E_1 is $\mathfrak{E}_V^0$–**independent** of E_2 iff $E_1 \not\Rightarrow_{\mathfrak{E}_V^0} E_2$. Also, if $\neg(E_1 \not\Rightarrow_{\mathfrak{E}_V^0} E_2)$, we simply write $E_1 \Rightarrow_{\mathfrak{E}_V^0} E_2$. Here, we say the E_1 is $\mathfrak{E}_V^0$–**dependent** of E_2 iff $E_1 \Rightarrow_{\mathfrak{E}_V^0} E_2$.*

Proposition 4.6. $\not\Rightarrow_{\mathfrak{E}_V^0}$ *is a preclusive relation.*

Proof. We recall that a preclusive relation is one which is irreflexive and symmetric. Let $E \in \mathfrak{E}^\circ{}_V$. Since $E \neq Id$, then by application of Proposition 4.2.3 (E, E) does not generate a unique $L_2 L_1$ operator and therefore $E \Rightarrow_{\mathfrak{E}_V^0} E$. Hence $\not\Rightarrow_{\mathfrak{E}_V^0}$ is irreflexive.

Now, suppose that $E_1, E_2 \in \mathfrak{E}^\circ{}_V$ are such that $E_1 \not\Rightarrow_{\mathfrak{E}_V^0} E_2$. Then (E_1, E_2) satisfies the conditions of Theorem 4.4. Since together, the four conditions of the theorem are symmetric (with conditions (i) and (ii) and conditions (iii) and (iv) being symmetric pairs), then (E_2, E_1) also satisfies the conditions of the theorem. Then by this theorem, we will have that $E_2 \not\Rightarrow_{\mathfrak{E}_V^0} E_1$. Hence, $\not\Rightarrow_{\mathfrak{E}_V^0}$ is symmetric. $\qquad \square$

Remark 4.4 From the previous proposition we can see that dependency relation $\Rightarrow_{\mathfrak{E}_V^0}$ is a similarity relation.

Proposition 4.7. *If $E_1 \Rightarrow E_2$ then $E_1 \Rightarrow_{\mathfrak{E}_V^0} E_2$.*

Proof. This follows from Corollary 4.3. $\qquad \square$

Proposition 4.8. *It is not the case that $E_1 \Rightarrow_{\mathfrak{E}_V^0} E_2$ implies that $E_1 \Rightarrow E_2$.*

Proof. In Example 4.2 we see (E_1, E_2) does not give a corresponding unique $L_2 L_1$ operator, hence $E_1 \Rightarrow_{\mathfrak{E}_V^0} E_2$ but $E_1 \not\Rightarrow E_2$. $\qquad \square$

Remark 4.5 From Proposition 4.7 and Proposition 4.8, we see that $\mathfrak{E}_V^0$–**dependency** is a more general notion of equivalence relation dependency that $\Rightarrow$ (or equivalently $\leq$). Similarly $\mathfrak{E}_V^0$–**independence** is a stricter notion of independence than $\not\Rightarrow$.

Theorem 4.5. *Let V be a finite set and E_1 and E_2 equivalence relations on V. Then $E_1 \nRightarrow_{\mathfrak{C}_V^0} E_2$ iff the following holds:*

(i) For each $[x]_{E_2}, [y]_{E_2} \in E_2$, if $[x]_{E_2} \neq [y]_{E_2}$ then $\boldsymbol{u}_{E_1}([x]_{E_2}) \neq \boldsymbol{u}_{E_1}([y]_{E_2})$,

(ii) For each $[x]_{E_1}, [y]_{E_1} \in E_1$, if $[x]_{E_1} \neq [y]_{E_1}$ then $\boldsymbol{u}_{E_2}([x]_{E_1}) \neq \boldsymbol{u}_{E_2}([y]_{E_1})$.

(iii) For any $[x]_{E_2} \in E_2$, there exists $[z]_{E_1} \in E_1$ such that, $|[x]_{E_2} \cap [z]_{E_1}| = 1$.

(iv) For any $[x]_{E_1} \in E_1$, there exists $[z]_{E_2} \in E_2$ such that, $|[x]_{E_1} \cap [z]_{E_2}| = 1$.

Proof. This follows directly from Theorem 4.4.

$\square$

4.3 Seeing One Equivalence Relation through Another

We will first give a proposition which will show a more explicit symmetry between conditions (i) and (ii) and conditions (iii) and (iv) in Theorem 4.4 for unique solutions.

Proposition 4.9. *Let V be a finite set and let E_1 and E_2 be two equivalence relations on V. Then;*

For any $[x]_{E_1} \in E_1$, $\exists [z]_{E_2} \in E_2$ such that, $|[x]_{E_1} \cap [z]_{E_2}| = 1$ iff it is not the case that $\exists Y, Z \in \mathcal{P}(V)$ such that $[x]_{E_1} = Y \cup Z$, $Y \cap Z = \emptyset$ and $\boldsymbol{u}_{E_2}(Y) = \boldsymbol{u}_{E_2}(Z) = \boldsymbol{u}_{E_2}([x]_{E_1})$.

Proof. We prove $\Rightarrow$ first. Let $[x]_{E_1} \in E_1$ and suppose that $\exists [z]_{E_2} \in E_2$ such that, $|[x]_{E_1} \cap [z]_{E_2}| = 1$. Then let $[x]_{E_1} \cap [z]_{E_2} = t$. Now for any spilt of $[x]_{E_1}$, that is for any $Y, Z \in \mathcal{P}(V)$ such that $[x]_{E_2} = Y \cup Z$ and $Y \cap Z = \emptyset$, t is in exactly one of these sets. Thus exactly one of $\boldsymbol{u}_{E_2}(Y)$, $\boldsymbol{u}_{E_2}(Z)$ contains $[t]_{E_2} = [z]_{E_2}$. Hence $\boldsymbol{u}_{E_2}(Y) \neq \boldsymbol{u}_{E_2}(Z)$.

We prove the converse by the contrapositive. Let $[x]_{E_1} \in E_1$ be such that for all $[z]_{E_2} \in E_2$ whenever $[x]_{E_1} \cap [z]_{E_2} \neq \emptyset$ (and clearly some such $[z]_{E_2}$ must exist), we have that $|[x]_{E_1} \cap [z]_{E_2}| \geq 2$. Suppose that $[x]_{E_1}$ has non-empty intersection with with n sets in E_2. We note that $n \geq 1$. Form a sequence of these sets; $R_1, \ldots R_n$. Since $|[x]_{E_1} \cap R_i| \geq 2$ for each i such that $i = 1, \ldots n$, let $\{a_{i1}, a_{i2}\}$ be in $[x]_{E_1} \cap R_i$ for each i such that $i = 1, \ldots n$. Let $Y = \{a_{i1} \mid i = 1, \ldots n\}$ and let $Z = [x]_{E_1} - Y$. Then, $[x]_{E_1} = Y \cup Z$, $Y \cap Z = \emptyset$ and $\boldsymbol{u}_{E_2}(Y) = \boldsymbol{u}_{E_2}(Z) = \boldsymbol{u}_{E_2}([x]_{E_1})$. $\square$

Using the preceding proposition we obtain an equivalent form of Theorem 4.4.

Theorem 4.6. *Let V be a finite set and E_1 and E_2 equivalence relations on V. Then (E_1, E_2) produces a unique $L_2 L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ operator iff the following holds:*

(i) For each $[x]_{E_2}, [y]_{E_2} \in E_2$, if $[x]_{E_2} \neq [y]_{E_2}$ then $\boldsymbol{u}_{E_1}([x]_{E_2}) \neq \boldsymbol{u}_{E_1}([y]_{E_2})$

(ii) For each $[x]_{E_1}, [y]_{E_1} \in E_1$, if $[x]_{E_1} \neq [y]_{E_1}$ then $\boldsymbol{u}_{E_2}([x]_{E_1}) \neq \boldsymbol{u}_{E_2}([y]_{E_1})$

(iii) For any $[x]_{E_2} \in E_2$, if $\exists Y, Z \in \mathcal{P}(V)$ such that $[x]_{E_2} = Y \cup Z$ and $Y \cap Z = \emptyset$ then $\boldsymbol{u}_{E_1}(Y) \neq \boldsymbol{u}_{E_1}(Z)$

(iv) For any $[x]_{E_1} \in E_1$, if $\exists Y, Z \in \mathcal{P}(V)$ if $[x]_{E_1} = Y \cup Z$ and $Y \cap Z = \emptyset$ then $\boldsymbol{u}_{E_2}(Y) \neq \boldsymbol{u}_{E_2}(Z)$

4.3.1 Conceptual Translation of the Uniqueness Theorem

The conditions of the above theorem can be viewed conceptually as follows: (i) Through the eyes of E_1, no two equivalence classes of E_2 are the same; (ii) Through the eyes of E_2, no two equivalence classes of E_1 are the same; (iii) No equivalence class in E_2 can be broken down into two smaller equivalence classes which are equal to it through the eyes of E_1; (iv) No equivalence class in E_1 can be broken down into two smaller equivalence classes which are equal to it through the eyes of E_2. In other words we view set $V \mathbf{mod}\, E_1$. That is, let $V \mathbf{mod} E_1$ be the set obtained from V after renaming the elements of V with fixed representatives of their respective equivalence classes in E_1. Similarly let $V \mathbf{mod} E_2$ be the set obtained from V after renaming the elements of V with fixed representatives of their respective equivalence classes in E_2. We then have the following equivalent conceptual version of Theorem 4.4

Theorem 4.7. *Let V be a finite set and E_1 and E_2 equivalence relations on V. Then (E_1, E_2) generate a unique $L_2 L_1$ operator iff the following holds:*

(i) No two distinct members of E_2 are equivalent in $V \mathbf{mod} E_1$.

(ii) No two distinct members of E_1 are equivalent in $V \mathbf{mod} E_2$.

(iii) No member E_2 can be broken down into two smaller sets which are equivalent to it in $V \mathbf{mod} E_1$.

(iv) No member E_1 can be broken down into two smaller sets which are equivalent to it in $V \mathbf{mod} E_2$.

5 Decomposing $U_2 U_1$ Approximations

We now investigate the case of double upper approximations. This is dually related to the case of double lower approximations because of the relationship between upper and lower approximations by the equation, $U(X) = -L(-X)$ (see property 10 in Section 2.1.1). The following proposition shows that the problem of finding solutions for this case reduces to the case in the previous section:

Proposition 5.1. *Let V be a finite set and let $U_2 U_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given fully defined operator on $\mathcal{P}(V)$. Then any solution (E_1, E_2), is also a solution of $L_2 L_1 : \mathcal{P}(V) \to P\mathcal{P}(V)$ operator where $L_2 L_1(X) = -U_2 U_1(-X)$ for any $X \in \mathcal{P}(V)$. Therefore, the solution (E_1, E_2) for the defined $U_2 U_1$ operator is a unique iff the solution for the corresponding $L_2 L_1$ operator is unique.*

Proof. Recall that $L_2 L_1(X) = -U_2 U_1(-X)$. Hence, if there exists a solution (E_1, E_2) which corresponds to the given $U_2 U_1$ operator, this solution corresponds to a solution for the $L_2 L_1$ operator which is based on the same (E_1, E_2) by the equation $L_2 L_1(X) = -U_2 U_1(-X)$. Similarly for the converse. $\qquad \square$

Algorithm: Let V be a finite set and let $U_2 U_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given fully defined operator on $\mathcal{P}(V)$. To solve for a solution, change it to solving for a solution for the corresponding $L_2 L_1$ operator by the equation $L_2 L_1(X) = -U_2 U_1(-X)$. Then, when we want to know the $L_2 L_1$ output of a set we look at the $U_2 U_1$ output of its complement set and take the complement of that. Next, use Algorithm 4.2 and the solution found will also be a solution for the initial $U_2 U_1$ operator.

5.1 Characterising Unique Solutions

Theorem 5.1. *Let V be a finite set and let $U_2U_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given fully defined operator on $\mathcal{P}(V)$. If (E_1, E_2) is a solution then, it is unique iff the following holds:*

(i) for each $[x]_{E_2}, [y]_{E_2} \in E_2$, if $[x]_{E_2} \neq [y]_{E_2}$ then $\boldsymbol{u}_{E_1}([x]_{E_2}) \neq \boldsymbol{u}_{E_1}([y]_{E_2})$,

(ii) for each $[x]_{E_1}, [y]_{E_1} \in E_1$, if $[x]_{E_1} \neq [y]]_{E_1}$ then $\boldsymbol{u}_{E_2}([x]_{E_1}) \neq \boldsymbol{u}_{E_2}([y]_{E_1})$.

(iii) For any $[x]_{E_2} \in E_2$, there exists $[z]_{E_1} \in E_1$ such that, $|[x]_{E_2} \cap [z]_{E_1}| = 1$.

(iv) For any $[x]_{E_1} \in E_1$, there exists $[z]_{E_2} \in E_2$ such that, $|[x]_{E_1} \cap [z]_{E_2}| = 1$.

Proof. This follows from Proposition 5.1 using Theorem 4.4. □

6 Decomposing U_2L_1 Approximations

For this case, we observe that $U_2L_1(X) = -L_2(-L_1(X)) = U_2(-U_1(-X))$. Since we cannot get rid of the minus sign between the Ls (or Us), duality will not save us the work of further proof here like it did in the previous section. In this section, we will see that U_2L_1 approximations are tighter than L_2L_1 (or U_2U_1) approximations. For this decomposition we will use an algorithm that is very similar to Algorithm 4.1, however notice the difference in step 2 where it only requires the use of minimal sets with respect to $\subseteq$ instead of minimum sets (which may not necessarily exist).

Algorithm 4.2: For Partial Decomposition of Double Successive Lower Approximations

Let V be a finite set. Given an input of a fully defined operator $U_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$, if a solution exists, we can produce a solution (S, R), i.e. where L_1 and U_2 are the lower and upper approximation operators of equivalence relations S and R respectively, by performing the following steps:

1. Let J be the set of output sets of the given U_2L_1 operator. We form the relation R to be such that for $a, b \in V$, $a \sim_R b \iff (a \in X \iff b \in X)$ for any $X \in J$. It is clear that R is an equivalence relation.

2. For each $Y \neq \emptyset$ output set, find the minimal pre-image sets with respect to $\subseteq$, Y_m, such that $U_2L_1(Y_m) = Y$. Collect all these minimal sets in a set K. Note that we can always find these minimal sets since $\mathcal{P}(V)$ is finite.

3. Using K, we form the relation S to be such that for $a, b \in V$, $a \sim_S b \iff (a \in X \iff b \in X)$ for any $X \in K$. It is clear that S is an equivalence relation.

4. Form the operator $U_RL_S : \mathcal{P}(V) \to \mathcal{P}(V)$ generated by (S, R). If for all $X \in \mathcal{P}(V)$, the given U_2L_1 operator is such that $U_2L_1(X) = U_RL_S(X)$, then (S, R) is a solution proving that a solution exists (note that it is not necessarily unique). Return (S, R). Otherwise, discard S and R and return 0 signifying that no solution exists.

We will prove the claim in step 4 in this section.

Lemma 6.1. *Let V be a set and $U_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given fully defined operator on $\mathcal{P}(V)$ with L_1 and E_2 based on unknown E_1 and E_2 respectively. Let R and S be equivalence relations defined on V as constructed in Algorithm 4.3. Then $E_2 \leq R$ and $E_1 = S$.*

Proof. We first prove $E_2 \leq R$. Now the output set of a non-empty set in $\mathcal{P}(V)$ is obtained by first applying the lower approximation L_1 to it and and after applying the upper approximation, U_2 to it. Hence by definition of U_2, the non-empty output sets are unions of equivalence classes of the equivalence relation which corresponds to U_2. If a is in an output set but b is not in it then they cannot belong to the same equivalence class of E_2 i.e. $a \not\sim_R b$ implies that $a \not\sim_{E_2} b$. Hence $E_2 \leq R$.

Now, the minimal pre-image, X say, of a non-empty output set which is a union of equivalence classes in E_2, has to be a union of equivalence classes in E_1. For suppose it was not. Let $Y = \{y \in X \mid [y]_{E_1} \not\subseteq X\}$. By assumption, $Y \neq \emptyset$. Then $L_1(X) = L_1(X - Y)$. Hence $U_2L_1(X) = U_2L_1(X - Y)$ but $|X - Y| < |X|$ contradicting minimality of X. Therefore, if a belongs to the minimal pre-image of a non-empty output set but b does not belong to it, then a and b cannot belong to the same equivalence class in E_1 i.e. $a \not\sim_S b$ which implies that $a \not\sim_{E_1} b$. Hence $E_1 \leq S$.

We now prove the converse, that $S \leq E_1$. For suppose it was not. That is, $E_1 < S$. Then there exists at least one equivalence class in S which is split into smaller equivalence classes in E_1. Call this equivalence class $[a]_S$. Then there exists $w, t \in V$ such that $[w]_{E_1} \subset [a]_S$ and $[t]_{E_1} \subset [a]_S$. Now consider the pre-images of a minimal output sets of U_2L_1, containing t. That is, X such that $U_2L_1(X) = Y$ where Y is the minimal output set such that $t \in Y$ and for any $X_1 \subset X$, $U_2L_1(X_1) \neq Y$. The following is a very useful observation.

Claim: For any $v \in \boldsymbol{u}_{E_1}([y]_{E_2})$, $[v]_S$ is a minimal set such that $U_2L_1([v]_S) \supseteq [y]_{E_2}$. The above follows because 1) $U_2L_1([v]_S) \supseteq [y]_{E_2}$ since $v \in \boldsymbol{u}_{E_1}([y]_{E_2})$ and 2) For any $Z \subset [v]_S$, $U_2L_1(Z) = \emptyset$ since $L_1(Z) = \emptyset$.

Now for $U_2L_1(X)$ to contain t, then it must contain $[t]_{E_2}$. Hence by the previous claim, $X = [t]_S$ is such a minimal pre-image of a set containing t. If L_1 is based on S, then $X = [t]_S = [a]_S$. However, if L_1 is based on E_1, then $X = [a]_S$ is not such a minimal set because $X = [t]_{E_1}$ is such that $U_2L_1(X) = Y$ but $[t]_{E_1} \subset [a]_S$. Hence, $U_RL_S(X) \neq U_{E_2}L_{E_1}(X)$ for all $X \in \mathcal{P}(V)$ which is a contradiction to (E_1, E_2) also being a solution for the given U_2U_1 operator. Thus we have that $E_1 = S$. $\square$

Lemma 6.2. *Let V be a finite set and $U_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a fully defined operator. If there exists equivalence pair solutions to the operator (E_1, E_2) which is such that there exists $[x]_{E_2}, [y]_{E_2} \in E_2$, such that $[x]_{E_2} \neq [y]_{E_2}$ and $\boldsymbol{u}_{E_1}([x]_{E_2}) = \boldsymbol{u}_{E_1}([y]_{E_2})$, then there exists another solution, (E_1, H_2), where H_2 is an equivalence relation formed from E_2 by combining $[x]_{E_2}$ and $[y]_{E_2}$ and all other elements are as in E_2. That is, $[x]_{E_2} \cup [y]_{E_2} = [z] \in H_2$ and if $[w] \in E_2$ such that $[w] \neq [x]_{E_2}$ and $[w]_{E_2} \neq [y]_{E_2}$, then $[w] \in H_2$.*

Proof. Suppose that (E_1, E_2) is a solution of a given U_2L_1 operator and H_2 is as defined above. Now, $U_2L_1(X) = Y$ iff the union of E_1-equivalence classes in X intersects the equivalence classes of E_2 whose union is equal to Y. So, in the (E_1, H_2) solution, the only way that $U_{H_2}L_{E_1}(X)$ could be different from $U_{E_2}L_{E_1}(X)$(which is $= U_2L_1(X)$) is if there some equivalence class of E_1 which either intersects $[x]_{E_2}$ but not $[y]_{E_2}$ or intersects $[y]_{E_2}$ but not $[x]_{E_2}$. However, this is not the case since we have that $\boldsymbol{u}_{E_1}([x]_{E_2}) = \boldsymbol{u}_{E_1}([y]_{E_2})$. Hence, $U_{E_2}L_{E_1}(X) = U_{H_2}L_{E_1}(X)$ for all $X \in \mathcal{P}(V)$ and therefore if (E_1, E_2) is a solution to the given operator then so is (E_1, H_2). $\square$

Next, we prove the claim in step 4 of Algorithm 4.2.

Theorem 6.1. *Let V be a finite set and $U_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ a fully defined operator. If there exists an equivalence relation pair solution, then there exists a solution (E_1, E_2), which satisfies,*

(i) for each $[x]_{E_2}, [y]_{E_2} \in E_2$, if $[x]_{E_2} \neq [y]_{E_2}$ then $\boldsymbol{u}_{E_1}([x]_{E_2}) \neq \boldsymbol{u}_{E_1}([y]_{E_2})$,

Furthermore $E_1 = S$ and $E_2 = R$, where (S, R) are the relations obtained by applying Algorithm 4.2 to the given U_2L_1 operator.

Proof. Suppose that there exists a solution (C, D). Then by Lemma 6.1, $C = S$, where S is produced by Algorithm 4.2. If (S, D) satisfies condition (i) of the theorem then take $(E_1, E_2) = (C, D)$. Otherwise, use repeated applications of Lemma 6.2 until we obtain a solution, (S, E_2) which satisfies the condition of the theorem. Since $\mathcal{P}(V)$ is finite this occurs after a finite number of applications of the lemma. Moreover, by Lemma 6.1, $E_2 \leq R$.

Consider the minimal sets in the output list of the given U_2L_1 operator. It is clear that these sets are union of one or more equivalence classes of E_2. Let $[y]_{E_2} \in E_2$ then for any $v \in \boldsymbol{u}_{E_1}([y]_{E_2}))$, $U_2L_1([v]_S) \supseteq [y]_{E_2}$ (by the claim in Lemma 6.1).

Claim: (i) For any $[y]_{E_2} \neq [z]_{E_2} \in E_2$, there exists an output set, $U_2L_1(X)$ such that it contains at least of $[y]_{E_2}$ or $[z]_{E_2}$ both it does not contain both sets.

Suppose that $[y]_{E_2} \neq [z]_{E_2} \in E_2$. By the assumed condition of the theorem, then $\boldsymbol{u}_{E_1}([y]_{E_2}) \neq \boldsymbol{u}_{E_1}([z]_{E_2})$. Hence either (i) there exists $a \in V$ such that $a \in \boldsymbol{u}_{E_1}([y]_{E_2})$ and $a \notin \boldsymbol{u}_{E_1}([z]_{E_2})$ or (ii) there exists $a \in V$ such that $a \notin \boldsymbol{u}_{E_1}([y]_{E_2})$ and $a \in \boldsymbol{u}_{E_1}([z]_{E_2})$. Consider the first case. This implies that $[a]_S \cap [y]_{E_2} \neq \emptyset$ while $[a]_S \cap [z]_{E_2} = \emptyset$. Therefore, $U_2L_1([a]_S) \supseteq [y]_{E_2}$ but $U_2L_1([a]_S) \not\supseteq [z]_{E_2}$. Similarly, for the second case we will get that $U_2L_1([a]_S) \supseteq [z]_{E_2}$ but $U_2L_1([a]_S) \not\supseteq [y]_{E_2}$ and the claim is shown.

We recall that $a \sim_R b \iff (a \in X \iff b \in X)$ for each X in the range of the given U_2L_1. From the previous proposition we have that $E_2 \leq R$. From the above claim we see that if $[y]_{E_2} \neq [z]_{E_2}$ in E_2 then there is an output set that contains one of $[y]_{E_2}$ or $[z]_{E_2}$, but not the other. Hence, if $x \not\sim_{E_2} y$ then $x \not\sim_R y$. That is, $R \leq E_2$. Therefore we have that $R = E_2$. □

6.1 Characterising Unique Solutions

Theorem 6.2. *Let V be a finite set and let $U_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a fully defined successive approximation operator on $\mathcal{P}(V)$. If (S, R) is returned by Algorithm 4.1, then (S, R) is the unique solution of the operator iff the following holds:*

(i) For any $[x]_R \in R$, there exists $[z]_S \in S$ such that, $|[x]_R \cap [z]_S| = 1$.

Proof. We prove $\Leftarrow$ direction first. So assume the condition holds. Then by Theorem 6.1 if there is a unique solution, it is (S, R) produced by Algorithm 4.2. We note that by Lemma 6.1, any other solution, (E_1, E_2) to the given U_2L_1 operator must be such that $E_1 = S$ and $E_2 \leq R$.

So, suppose to get a contradiction, that there exists a solution (E_1, E_2) which is such that $E_2 < R$. That is, E_2 contains a splitting of at least one of the equivalences classes of R, say $[a]_R$. Hence $|[a]_R| \geq 2$. By assumption there exists a $[z]_S \in S$ such that $|[a]_R \cap [z]_S| = 1$. Call the element in this intersection v say. We note that $[v]_S = [z]_S$. Now as $[a]_R$ is spilt into smaller classes in E_2, v must be in one of these classes, $[v]_{E_2}$. Now, $U_2L_1([v]_S)$ when U_2 is based on E_2, contains $[v]_{E_2}$ but does not contain $[a]_R$. This is because $[v]_S \cap ([a]_R - [v]_{E_2}) = \emptyset$. That is, $U_{E_2}L_S([v]_S) \not\supseteq [a]_R$ but $U_RL_S([v]_S) \supseteq [a]_R$. Hence $U_{E_2}L_S(X) \neq U_RL_S(X)$ for all $X \in \mathcal{P}(V)$. This is a contradiction to

(S, E_2) also being a solution to the given U_2L_1 operator for which (S, R) is a solution. Hence we have a contradiction and so $E_2 = R$.

Now we prove $\Rightarrow$ direction. Suppose that (E_1, E_2) is the unique solution, and assume that the condition does not hold. By uniqueness, $(E_1, E_2) = (S, R)$. Then, there exists an $[x]_R \in R$ such that for all $[y]_S \in S$ such that $[x]_R \cap [y]_S \neq \emptyset$ we have that $|[x]_R \cap [y]_S| \geq 2$.

Suppose that $[x]_R$ has non-empty intersection with with n sets in S. We note that $n \geq 1$. Form a sequence of these sets; $S_1, \ldots S_n$. Since $|[x]_R \cap S_i| \geq 2$ for each i such that $i = 1, \ldots n$, let $\{a_{i1}, a_{i2}\}$ be in $[x]_R \cap S_i$ for each i such that $i = 1, \ldots n$. We split $[x]_R$ to form a finer E_2 as follows: Let $P = \{a_{i1} \mid i = 1, \ldots n\}$ and $Q = [x]_R - P$ be two equivalence classes in E_2 and for the rest of E_2, for any $[y]_R \in R$ such that $[y]_R \neq [x]_R$, let $[y] \in E_2$ iff $[y] \in R$. Now, $U_R L_S(X) = Y$ iff the union of S-equivalence classes in X intersects equivalence classes of E_2 whose union is equal to Y. So, for the (S, E_2) solution, the only way that $L_{E_2} L_S(X)$ could be different from $L_R L_S(X)$ is if there is an equivalence class in S which intersects P but not Q or Q but not P. However, this is not the case because $\boldsymbol{u}_S(P) = \boldsymbol{u}_S(Q)$. Hence, $L_R L_S(X) = L_{E_2} L_S(X)$ for all $X \in \mathcal{P}(V)$ and if (S, R) is a solution for the given vector, then so is (S, E_2) which is a contradiction of assumed uniqueness of (S, R). $\square$

The following result sums up the effects of Theorem 6.1 and Theorem 6.2.

Theorem 6.3. *Let V be a finite set and let $U_2 L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given fully defined operator on $\mathcal{P}(V)$. Then there exists a unique pair of equivalence relations solution (E_1, E_2) iff the following holds:*

(i) for each $[x]_{E_2}, [y]_{E_2} \in E_2$, if $[x]_{E_2} \neq [y]_{E_2}$ then $\boldsymbol{u}_{E_1}([x]_{E_2}) \neq \boldsymbol{u}_{E_1}([y]_{E_2})$,

(iii) For any $[x]_{E_2} \in E_2$, there exists $[z]_{E_1} \in E_1$ such that, $|[x]_{E_2} \cap [z]_{E_1}| = 1$.

7   Decomposing $L_2 U_1$ Approximations

For this case we observe that $L_2 U_1$ is dual to the case previously investigated $U_2 L_1$ operator. Due to the duality connection between $L_2 U_1$ and $U_2 L_1$, the question of unique solutions of the former reduces to the latter as the following proposition shows.

Proposition 7.1. *Let V be a finite set and let $L_2 U_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given fully defined operator on $\mathcal{P}(V)$. Then any solution (E_1, E_2), is also a solution of $U_2 L_1 : \mathcal{P}(V) \to P\mathcal{P}(V)$ operator where $U_2 L_1(X) = -L_2 U_1(-X)$ for any $X \in \mathcal{P}(V)$. Therefore, the solution (E_1, E_2) for the defined $U_2 U_1$ operator is a unique iff the solution for the corresponding $U_2 L_1$ operator is unique.*

Proof. Recall that $U_2 L_1(X) = -L_2 U_1(-X)$. Hence, if there exists a solution (E_1, E_2) which corresponds to the given $U_2 L_1$ operator, this solution corresponds to a solution for the $L_2 U_1$ operator which is based on the same (E_1, E_2) by the equation $L_2 U_1(X) = -U_2 L_1(-X)$. Similarly for the converse. $\square$

Algorithm: Let V be a finite set and let $L_2 U_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given fully defined operator on $\mathcal{P}(V)$. To solve for a solution, change it to solving for a solution for the corresponding $U_2 L_1$ operator by the equation $U_2 L_1(X) = -L_2 U_1(-X)$. Then, when we want to know the $U_2 L_1$ output of a set we look at the $L_2 U_1$ output of its complement set and take the complement of that. Next, use Algorithm 4.2 and the solution found will also be a solution for the initial $L_2 U_1$ operator.

7.1 Characterising Unique Solutions

Theorem 7.1. *Let V be a finite set and let $L_2U_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ be a given fully defined operator on $\mathcal{P}(V)$. If (E_1, E_2) is a solution, then it is unique iff the following holds:*

(i) for each $[x]_{E_2}, [y]_{E_2} \in E_2$, if $[x]_{E_2} \neq [y]_{E_2}$ then $\boldsymbol{u}_{E_1}([x]_{E_2}) \neq \boldsymbol{u}_{E_1}([y]_{E_2})$,
(ii) For any $[x]_{E_2} \in E_2$, there exists $[z]_{E_1} \in E_1$ such that, $|[x]_{E_2} \cap [z]_{E_1}| = 1$.

Proof. This follows from Proposition 7.1, Theorem 6.1 and Theorem 6.2. $\qquad\square$

8 Conclusion

We have defined and examined the consequences of double successive rough set approximations based on two, generally unequal equivalence relations on a finite set. We have given algorithms to decompose a given defined operator into constituent parts. Additionally, in sections 4.2 and 4.3 we have found a conceptual translation of the main results which is very much in the spirit of what Yao suggested in [34]. These type of links are especially helpful in forming a coherent map of the mass of existing literature out there.

This type of analysis can be seen as somewhat analogous to decomposing a wave into constituent sine and cosine waves using Fourier analysis. In our case, we work out the possibilities of what can be reconstructed if we know that a system has in-built layered approximations. It is possible that some heuristics of how the brain works can be modelled using such approximations and cognitive science is a possible application for the theory which we have begun to work out.

References

[1] T. Arciszewski and W Ziarko. Adaptive expert system for preliminary design of windbracings in steel skeleton structures. *Second Century of Skyscraper, Van Norstrand*, pages 847–855, 1987.

[2] T. Arciszewski and W Ziarko. Inductive learning in civil engineering: rough sets approach. *Microcomputers and Civil Engineering*, 5(1), 1990.

[3] Rafael Bello and JosÃľ Luis Verdegay. Knowledge engineering for rough sets based decision-making models. *International Journal of Intelligent Systems*, 29(9):823–835, 2014.

[4] G. Cattaneo and D. Ciucci. A quantitative analysis of preclusivity vs. similarity based rough approximations. *LNCS: Rough Sets and Current Trends in Computing*, 2475:69–76, 2002.

[5] J. S. Deogun, V. V. Raghavan, and H. Sever. Rough set based classification methods and extended decision tables. *Proc. of The Int. Workshop on Rough Sets and Soft Computing*, pages 302–309, 1994.

[6] J.W. Grzymala-Busse. Lers-a system for learning from examples based on rough sets. *Theory and Decision Library: Intelligent Decision Support*, 11:3–18, 1992.

[7] J.W. Grzymala-Busse, S.Y. Sedelow, and W.A. Sedelow Jr. *Rough Sets and Data Mining*, chapter Machine Learning and Knowledge Acquisition, Rough Sets, and the English Semantic Code. Springer US, 1997.

[8] A. Gupta, N. Kumar, and Bhatnagar V. Analysis of medical data using data mining and formal concept analysis. *World Academy of Science, Engineering and Technology*, 11:61–64, 2005.

[9] A. E. Hassanien, A. Abraham, J.F. Peters, and G Schaefer. Rough sets in medical informatics applications. *Advances in Intelligent and Soft Computing: Applications of Soft Computing*, 58:23–30, 2009.

[10] K.Y. Hu, Y.C. Lu, and C.Y. Shi. Feature ranking in rough sets. *AI Communications*, 16(1):41–50, 2003.

[11] X. Hu and N. Cercone. Learning in relational databases: A rough set approach. *Computational Intelligence*, 11(2):323–338, 1995.

[12] X-M. Huang and Y-H. Zhang. A new application of rough set to ecg recognition. *Int. Conference on Machine Learning and Cybernetics*, 3:1729–1734, 2003.

[13] S. Kobashi, K. Kondo, and Y. Hata. Rough sets based medical image segmentation with connectedness. *Int. Forum on Multimedia and Image Processing*, pages 197–202, 2004.

[14] Napoles G. Bello R. Mkrtchyan L. Depaire B. Vanhoof K. Leon, M. Tackling travel behaviour: An approach based on fuzzy cognitive maps. *International Journal of Computational Intelligence Systems*, 6:1012–1039, 2013.

[15] P. Mahajan, Kandwal R., and R. Vijay. Rough set approach in machine learning: A review. *International Journal of Computer Applications*, 56(10):1–13, 2012.

[16] G. Napoles, I. Grau, K. Vanhoof, and R. Bello. Hybrid model based on rough sets theory and fuzzy cognitive maps for decision-making. *LNCS Rough Sets and Intelligent Systems Paradigms*, 8537:169–178, 2014.

[17] P. Pagliani and M. Chakraborty. *A Geometry of Approximation Rough Set Theory: Logic, Algebra and Topology of Conceptual Patterns*. Trends in Logic (Book 27). Springer, 2008.

[18] Z. Pawlak. Rough sets. *International Journal of Parallel Programming*, pages 341–356, 1982.

[19] Z. Pawlak. *Rough Sets: Theoretical Aspects of Reasoning about Data*, volume 9 of *Theory and Decision Library D*. Springer Netherlands, 1991.

[20] Z. Pawlak. *Rough Sets: Theoretical Aspects of Reasoning about Data*, volume 9 of *Theory and Decision Library D:*. Springer Netherlands, 1 edition, 1991.

[21] Z. Pawlak. AI and intelligent industrial applications: The rough set perspective. *Cybernetics and Systems*, 31(3):227–252, 2000.

[22] Z. Pawlak. Rough sets, decision algorithms and bayes' theorem. *European Journal of Operational Research*, 136(1):181–189, 2002.

[23] G. Peters, P. Lingras, D. Slezak, and Y.Y. Yao, editors. *Rough Sets: Selected Methods and Applications in Management and Engineering*. Advanced Information and Knowledge Processing. Springer London, 2012.

[24] I. Pisica and P. Postolache. Rough set theory and its applications in electrical power engineering. a survey. *UPB Scientific Bulletin, Series C: Electrical Engineering*, 72(1), 2010.

[25] A.W Przybyszewski. The neurophysiological bases of cognitive computation using rough set theory. *LNCS Transactions on Rough Sets IX*, 5390:287–317, 2008.

[26] M. Salamã and M. Lãşpez-Sãnchez. Rough set based approaches to feature selection for case-based reasoning classifiers. *Pattern Recognition Letters*, 32(2):280–292, 2011.

[27] R. Slowinski, editor. *Intelligent Decision Support. Handbook of Applications and Advances of the Rough Set Theory.* Kluwer Academic Publishers, Boston, London, Dordrecht, 1992.

[28] J Stefanowski. *Rough Sets in Knowledge Discovery*, chapter On Rough Set Based Approaches to Induction of Decision Rules. Physica-Verlag, Heidelberg, 1998.

[29] R.W. Swiniarski. Rough set methods in feature reduction and classification. *International Journal of Applied Mathematics and Computer Science*, 11(3), 2001.

[30] B.K. Tripathy, D.P. Acharjya, and V. Cynthya. A framework for intelligent medical diagnosis using rough set with formal concept analysis. *International Journal of Artificial Intelligence and Applications*, 2(2):45–66, 2011.

[31] S. Tsumoto. Mining diagnostic rules from clinical databases using rough sets and medical diagnostic model. *Information Sciences*, 162(2), 2004.

[32] S. Tsumoto. Rough sets and medical differential diagnosis. *Intelligent Sysytems Reference Library: Rough Sets and Intelligent Sysytems - Professor Zdzislaw Pawlak in Memoriam*, 42:605–621, 2013.

[33] S. Vluymans, L. DâĂŹeer, Y. Saeys, and C. Cornelis. Applications of fuzzy rough set theory in machine learning: a survey. *Fundamenta Informaticae*, 142(1-4):53–86, 2015.

[34] Y.Y. Yao. The two sides of the theory of rough sets. *Knowledge-Based Systems*, 80:67–77, 2015.

[35] M. Yorek and S.. Narli. Modeling of cognitive structure of uncertain scientific conceptsers using fuzzy-rough sets and intuitionistic fuzzy sets: Example of the life concept. *International Journal of Uncertainty Fuzziness and Knowledge-Based Systems*, 17(5):747–769, year.

[36] N. Zhong, J.Z. Dong, and S. Ohsuga. Using rough sets with heuristics for feature selection. *Journal of Intelligent Information Systems*, 16:199–214, 2001.

[37] L. Zhou and F. Jiang. A rough set approach to feature selection based on relative decision entropy. *LNCS Rough Sets and Knowledge Technology*, 6954:110–119, 2011.