On Relational Extensions and Successive Approximation Operators of Rough Set Theory

by

Alexa Gopaulsingh

Submitted to Central European University

In partial fulfillment of the requirements for the degree of Doctor of Philosophy in Mathematics and its Applications

Supervisor: László Csirmaz

CENTRAL EUROPEAN UNIVERSITY

Budapest, Hungary

2016

I, the undersigned [Alexa Gopaulsingh], candidate for the degree of Doctor of Philosophy in Mathematics and its Applications at the Central European University, Mathematics and its Applications, declare herewith that the present thesis is exclusively my own work, based on my research and contains only such external information as properly credited in notes and bibliography. I declare that no unidentified and illegitimate use was made of the work of others, and no part of this thesis infringes on any person's or institution's copyright. I also declare that no part of this thesis has been submitted in this form to any other institution of higher education for an academic degree.

Budapest, 15 November 2017

Signature

Acknowledgements

I would like to deeply thank all of my family. My parents, ***Dawne Gopaulsingh*** *and* ***Ian Gopaulsingh*** *who has always fully encouraged and supported me. My brothers* ***Jonathan Gopaulsingh*** *and* ***Adrian Gopaulsingh*** *who gave me the very quick computer which I did this thesis on. I would also like to express my gratefulness for my grandmother* ***Lucille Ali*** *and my beloved late aunt,* ***Theresa Lygeia Ali****. Without you all I would not be who I am or where I am today.*

I would also like to thank the wider circle of friends and people who have helped, in some form or another, along the way.

I am greatly indebted to my former professor , Prof. ***E.J. Farrell****, from whom I learnt so much. I would also like very much like to thank my country, Trinidad and Tobago for funding me.*

I am also very grateful to my current supervisor, Prof. ***László Csirmaz*** *for his wisdom and patience and for the innumerable things I learnt from him as well.*

Last but not least, I would like to thank my Hungarian family. I am very grateful for my mother-in-law, ***Judit Környei*** *, who treats me like her own daughter. To my very loving and supportive husband,* ***Marcell****, and to my wonderful and amazing little daughter,* ***Penni****, I would like to express my deep love and thankfulness. You guys have brought such joy and happiness into my life and I thank you very much for your enthusiastic support and love during my research and in general.*

To Aunty Lygeia...

Abstract

We investigate non-dual relational generalisations of rough sets and find a generalisation which satisfies many nice properties. Additionally, we work out some consequences of relativised indistinguishability using graphs. Lastly, we consider successive double approximations, $L_2L_1,\ U_2U_1, U_2L_1, L_2U_1$ based on two equivalence relations on a set V. We consider the case of these operators being given defined on $\mathcal{P}(V)$ and ask if we can reconstruct the equivalence relations which they may be based on. Directly related to this, is the question of when there are unique solutions to a given defined operator and the existence of conditions which may characterise this case. We find and prove these characterising conditions that equivalence relation pairs should satisfy in order to generate unique operators.

Table of Contents

Chapter 1

Introduction

Rough Set Theory was introduced by Pawlak in the early 1980s, see [59], partially motivated by the desire to organise medical databases, see [53]. Most areas and regions of mathematics, if used as tools for modelling real-world data, are better suited to model data which fits precisely into some definition or categories. Exceptions to this are probabilistic modelling, fuzzy modelling and modelling based on many-valued logics. Rough set theory has been relatively recently added to this list. Hence, in hindsight it is perhaps not so surprising that rough set theory has natural connections to probability theory [97, 102, 109, 88], fuzzy sets [60, 90, 92, 47, 68, 26] and many-valued logics [5, 25, 6, 57, 21] and indeed much more.

Rough set approximations explicitly takes into account that some objects under one's consideration may not neatly fit into the categories which one may be using. In this case, we try to give the best categorical approximations to the objects in the form of two approximations: a lower approximation and an upper approximation. Conceptually, the idea may be viewed as this: Given an object which does not fit into the template/structures which we can see, which is the closest structure(s) to it? Perhaps the largest structure smaller than it or the smallest structure larger than it if these exist and we have a coherent notion of larger and smaller which satisfies a certain order type. This is related to the way of viewing it in [98] where definability is given as a primitive notion and then the notions of rough set approximations fall out naturally from it. Here, we may consider a set theoretical set-up and a definable set of sets which obey some closure operations. Then the lower approximation of a given set is the largest definable set contained in it and the upper approximation is the smallest definable set which it is contained in. These approximations also very naturally relate to the topological notions of an interior and closure of a set under some topology. Hence, this motivates many natural topological connections and generalisations [46, 42, 89, 72, 106, 70]. Rough sets also provide an excellent modelling tool for vagueness, see [15, 13, 87, 8, 75] and relatedly there are rough set extensions via similarity relations see [2, 1, 64, 91, 40].

Rough set formalisation usually includes a set equipped with an equivalence relation defined on it. Elements in an equivalence class are seen as indistinguishable. Informally speaking, under this interpretation, these elements should always 'occur' together. Hence, when presented with a subset of the domain one can say that the elements whose equivalence classes are contained in the set are surely in the set while elements for which their equivalence class is only partly contained in the set, are possibly in the set (also elements whose equivalence class are completely outside of the set are surely not contained in the set). This motivates a modal interpretation of rough sets and thus there are many studies in this direction, see [100, 80, 85, 51, 45].

There are different but equivalent ways to define rough set approximations which lead to non-equivalent rough set extensions. In [96], Yao gives three of these definitions. They are; an element based definition, a granule based definition and a subsystem based definition. Let V be a finite set and E be an equivalence relation on V. For $x \in V$, let $[x]_E$ be an equivalence class of x in E. Also, let $\sigma(E)$ be the set containing the equivalence classes of E and the empty set which is closed under taking unions. Then, for $X \subseteq V$, the lower approximation and upper approximation of X with respect to E are denoted by $\boldsymbol{l}_E(X)$ and $\boldsymbol{u}_E(X)$ respectively. For the element based definition, an element v is in $\boldsymbol{l}_E(X)$ iff all elements in its equivalence class are in X and v is in $\boldsymbol{u}_E(X)$ iff at least one of the elements in its equivalence class is in X. In the granule based definition, $\boldsymbol{l}_E(X)$ is the union of equivalence classes of elements whose equivalence classes are contained in X while $\boldsymbol{u}_E(X)$ is the union of equivalence classes which has non-empty intersection with

X. In the subsystem based definition, $\boldsymbol{l}_E(X)$ is the largest set in $\sigma(E)$ which is contained in X, while $\boldsymbol{u}_E(X)$ is the smallest set in $\sigma(E)$ which contains X. The element based definition inspires modal extensions, the granule based definition inspires extensions in granular computing see [54, 49, 48, 11, 81] and the subsystem definition inspires extensions in closure systems and algebra, see [19, 18, 9, 7, 67]. In [16], Cattaneo discusses what conditions should be met for generalised operators to rightly be considered upper and lower approximations and he compares the tightness of different generalisations. In [28], Düntsch showed that rough set approximations form a regular double Stone algebra.

Hence we see that the perspectives of different definitions of rough set approximations motivates many different extensions and directions. This shows the richness of the rough set formulation in that it has non-trivial interpretations in so many areas of mathematics. Therefore, it is conceptually very general yet has enough non-trivial content to be useful. Thus it can provide a unifying perspective on some diverse areas of mathematics. It is also related to left and right adjoints of Galois connections, see [71]. More generally in this direction, there has also been studies of rough sets in category theory, see [44, 50]. This is not to say that rough set theory is without foundational issues. In [10], Chakraborty and Banerjee discuss issues about rough sets with respect to language dependency and the problem of the referent and the background/context within which rough set approximations are defined.

In the above preface, we have touched upon the theoretical formulation of rough set theory but it also has tremendous practical applications. This comes from the calculation of reducts and decision rules for data. The data is mined to extract decision rules of manageable size (i.e. attribute reduction) so predictions can be made. It has been argued that rough set theory can be used to make decisions on the data in the absence of major prior assumptions in [63]. From this perspective, it is perhaps not so surprising that this leads to an explosion of applications. Hence rough set analysis adds to the tools of Bayes' Theorem and regression analysis for feature selection and pattern recognition in data mining [**?**, 103, 37, 104, 79, 27, 73, 78]. Applications include in medical databases [84, 83, 82, 35, 36, 39, 41], artificial intelligence and machine learning [52, 38, 34, 33, 86, 62, 76], engineering [3, 4, 12, 66, 65] and cognitive science [56, 43, 69, 58, 101]. In fact, it has been argued by Yao in [99] that there is an imbalance in the literature of rough sets between the conceptual development of the theory and the practical computational development. Here, he claims that the computational literature now far outweigh the conceptual theoretical literature and that it would be useful for the field if this imbalance were somewhat corrected. He started his suggestion in [99] where he gave a conceptual example of reducts which unify three different looking reduct definitions used in the literature. We agree that more work of this type would be helpful in organising and making a more coherent map of the huge mass of rough set literature which is present. Partly for this reason and partly to aid motivation, a conceptual translation or possible interpretation of results are sometimes provided, for example in Section 4.2.3 in Chapter 4.

This thesis is separated into three main parts. The first part in Chapter 2 deals with relational extensions. There are many such investigations in the literature, see [100, 31, 108, 29, 105, 107, 95, 93]. In [32], they define rough set approximations outlining the possibilities of using successor and predecessor sets. Here, we formulate non-dual relational extensions which uses successor and predecessor sets together and work out features of this definition and compare it to a standard definition given in the literature. We then apply this definition to answer a question posed in [74]. At the end of this chapter, we document a special non-dual extension of rough sets which interestingly satisfies almost all of the usual rough set properties except for duality.

In Chapter 3, we investigate different indistinguishability notions. We extend work from [22] which considers rough set approximations on graphs. Essentially, one can consider a graph as a relation on the nodes and one can form an equivalence relation on the nodes from this relation based on indistinguishability via the relation. Moreover, one can form relativised indistinguishability equivalence relations with respect to subsets of the nodes instead of the full node set. In [22], Chiaselotti et al formulate this setup and here we develop some consequences of it. We compare the indistinguishability of nodes of a graph between different relativised indistinguishability relations. Furthermore, we briefly extend the formulation to the hypergraph setting and provide an application via a database interpretation which suggests that further studies in this direction may be worthwhile. Lastly, we found a nice equivalence to Cantor's Diagonal Theorem which is related to indistinguishability and discuss the possible related effects of vagueness on models.

In Chapter 4, we will consider approximating a set successively with two, in general different, equivalence relations. Let V be a set and E_1 and E_2 be equivalence relations on V with L_1, U_1 and L_2, U_2 being their respective upper and lower approximations. Let $X \subseteq V$, we will consider the cases of X first being approximated by L_1 and then L_2, denoting this operation by $L_2L_1(X)$. We consider the setup of having the outputs of all the elements of $\mathcal{P}(V)$ and then the question becomes, can we decompose E_1 and E_2 from it? Will it have a unique (E_1, E_2) solution or will other pairs of equivalence relations produce the same output of L_2L_1 on $\mathcal{P}(V)$? We will find that the answer is that in some cases it does and in some cases it does not. So the next question becomes, are there conditions which characterise pairs of equivalence relations which give unique solutions? We will find these conditions and prove the characterisation. Next, we notice that unique pairs give rise to a preclusive relation. In [17], Cattaneo and Ciucci found that preclusive relations are quite useful for using rough approximations in information systems. This will lead us to define a related notion of independence of equivalence relations from it. Then, we will prove equivalent conditions of two of the four characterising conditions which are more conceptual. The first version of those conditions, while less illuminating, are easier to use in practice. This will lead us to an elegant conceptual translation of all the conditions. Lastly, we will consider the cases of the remaining operators, U_2U_1, U_2L_1 and L_2U_1. We note that the L_2L_1 and U_2U_1 cases are dual to each other and similarly for the U_2L_1 and L_2U_1 cases.

Chapter 2

Relational Extensions of Rough Sets

2.1 Relational Generalisations

Given the definition of rough sets, it is natural to ask what happens if we relax the equivalence relation to an arbitrary relation or other special relation. This has been well studied in the literature, see [100, 31, 108, 29, 105, 107, 95, 93]. In Section 2.2 we will recall a standard relational generalisation and compare properties which special relations satisfy under this generalisation. We will use a table similar to one in [74] and from it, one can obtain three characterising properties of reflexive relations, symmetric relations and transitive relations each. In [93], Yao did work which, though not directly stated, essentially implies these three propositions using operator theory. So, here we mention them so they can be seen directly in this simple form and also for comparison purposes with Section 2.3. Then we give brief, direct proofs which does not use operator terminology.

Next, we will observe that when generalising from rough sets based on equivalence relations to general relations, in the literature one usually sees either predecessor sets for both approximations or more commonly successor sets for both approximations. The approximation properties with predecessor and successor sets has been combinatorically mentioned in [32] and well as [95]. In [95], Yao also groups lower and upper approximations together which uses both predecessor sets each or successor sets each or which uses both each. He made these pairs most probably because they form dual operators as such. In Section 2.2, we show that if we are willing to give up duality, and combine a lower approximation which uses successor sets and an upper approximation which uses predecessor sets, we can find a generalisation which pretty much satisfies everything else except for duality, for pre-order (reflexive and transitive) relations. Here too, we construct a table and find a characterising property of transitive relations. From comparison between the two tables, we will see that none of the rough set generalisations along the special relations examined in the first table, satisfies so many properties as the pre-order generalisation of the non-dual extension in the second table. We believe that this makes this form of generalisation, and more specially the pre-order case of it, worthy of further consideration. In this direction, we will give two uses of this generalisation. One is that a covering operator mentioned in [74] is a special case of this operator, which explains why Samanta and Chakraborty noticed that it satisfied so many rough set properties and they remarked that it should be investigated further. Here, we see the reason for that well-behaved operator by placing it in this more general context. Lastly we, will give an example of a logical interpretation of this operator which shows that results about it can be nicely transferred to other areas.

2.2 Pawlak's Rough Sets

We recall some definitions and basic notions of rough sets which can be found in [59]. Let V be a finite non-empty set and E be an equivalence relation on V. Let V/E denote the set of equivalence classes of E. A set $X \subseteq V$ is said to be *E-exact* if it is equal to a union of some of the equivalence classes of E. If X cannot be represented in this way, it is said to be *E-inexact* or *E-rough* or simply *rough* if the equivalence relation under consideration is clear. In this case, we may approximate it with two exact sets, the lower and upper approximations respectively as defined below:

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

$$\boldsymbol{u}_E(X) = \{x \in V \mid [x]_E \cap X \neq \emptyset\}. \tag{2.1}$$

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

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

$$\boldsymbol{u}_E(X) = \bigcup\{Y \in V/E \mid Y \cap X \neq \emptyset\}. \tag{2.2}$$

The pair (V, E) is known as an *approximation space.*

Many times, several equivalence relations are considered over on set. A *knowledge base*, $K = (V, \mathcal{E})$ is defined with $\mathcal{E}$ being a family of equivalence relation over V. If $\mathcal{P} \subseteq \mathcal{E}$, the $\bigcap \mathcal{P}$ is an equivalence relation as well. The intersection of all equivalence relations belonging to $\mathcal{P}$ is denoted by $IND(\mathcal{P}) = \bigcap \mathcal{P}$. This is known as 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 [61] some definitions about different types of roughly definable and undefinable sets. Let V be a set then for $X \subseteq V$:

(i) If $\boldsymbol{l}_E(X) \neq \emptyset$ and $\boldsymbol{u}_E(X) \neq V$, then X is called *roughly E-definable.*

(ii) If $\boldsymbol{l}_E(X) = \emptyset$ and $\boldsymbol{u}_E(X) \neq V$, then X is called *internally roughly E-undefinable.*

(iii) If $\boldsymbol{l}_E(X) \neq \emptyset$ and $\boldsymbol{u}_E(X) = V$, then X is called *externally roughly E-definable.*

(iv) If $\boldsymbol{l}_E(X) = \emptyset$ and $\boldsymbol{u}_E(X) = V$, then X is called *totally roughly E-definable.*

2.2.1 List of properties satisfied by Rough Sets based on Equivalence Relations

In Pawlak's book, see [61], he lists these properties of rough sets based on equivalence relations which we repeat here. Let V be the domain of discourse and $X, Y \subseteq V$. Then, the following holds:

1)$\boldsymbol{l}_E(X) \subseteq X \subseteq \boldsymbol{u}_E(X)$,

2)$\boldsymbol{l}_E(\emptyset) = \boldsymbol{u}_E(\emptyset) = \emptyset; \quad \boldsymbol{l}_E(V) = \boldsymbol{u}_E(V) = V$,

3)$\boldsymbol{u}_E(X \cup Y) = \boldsymbol{u}_E(X) \cup \boldsymbol{u}_E(Y)$,

4)$\boldsymbol{l}_E(X \cap Y) = \boldsymbol{l}_E(X) \cap \boldsymbol{l}_E(Y)$,

5)$X \subseteq Y \Rightarrow \boldsymbol{l}_E(X) \subseteq \boldsymbol{l}_E(Y)$,

6)$X \subseteq Y \Rightarrow \boldsymbol{u}_E(X) \subseteq \boldsymbol{u}_E(Y)$,

7)$\boldsymbol{l}_E(X \cup Y) \supseteq \boldsymbol{l}_E(X) \cup \boldsymbol{l}_E(Y)$,

8)$\boldsymbol{u}_E(X \cap Y) \supseteq \boldsymbol{u}_E(X) \cap \boldsymbol{u}_E(Y)$,

9)$\boldsymbol{l}_E(-X) = -\boldsymbol{u}_E(X)$,

10)$\boldsymbol{u}_E(-X) = -\boldsymbol{l}_E(X)$,

11)$\boldsymbol{l}_E(\boldsymbol{l}_E(X)) = \boldsymbol{u}_E(\boldsymbol{l}_E(X)) = \boldsymbol{l}_E(X)$,

12)$\boldsymbol{u}_E(\boldsymbol{u}_E(X)) = \boldsymbol{l}_E(\boldsymbol{u}_E(X)) = \boldsymbol{u}_E(X)$.

2.2.2 Reducts, Core, Dependencies and Decision Rules

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 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 2.2.1. *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 $\mathcal{C}$ and $\mathcal{D}$ respectively. We then use these to make predictions of 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 condition 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.

The following definition compares how much one equivalence relation is consistent with another.

Definition 2.2.2. *Let V be a finite non-empty set and let C and D be equivalence relations on V. The* positive region *of the partition D with respect to C is given by,*

$$POS_C(D) = \bigcup_{X \in D} \boldsymbol{l}_C(X), \tag{2.3}$$

Definition 2.2.3. *We say the D* depends on C in a degree k, *where $0 \leq k \leq 1$, denoted by $C \Rightarrow_k D$, if*

$$k = \gamma(C, D) = \frac{|POS_C(D)|}{|V|}. \tag{2.4}$$

When $k = 1$, we simply note that C depends totally on D i.e $C \Rightarrow D$.

Definition 2.2.4. *Let $IND(\mathcal{C}) = \bigcap_{C \in \mathcal{C}} V/C$ and $IND(\mathcal{D}) = \bigcap_{D \in \mathcal{D}} V/D$ where, $\mathcal{C}$ and $\mathcal{D}$ are sets of decision and condition attributes respectively. We say that $\mathcal{C}_1 \subseteq \mathcal{C}$ is a $\mathcal{D}$-reduct of $\mathcal{C}$ if $\mathcal{C}_1$ is a minimal subset of $\mathcal{C}$ such that,*

$$\gamma(IND(\mathcal{C}), IND(\mathcal{D})) = \gamma(IND(\mathcal{C}_1), IND(\mathcal{D})) \tag{2.5}$$

where $IND(\mathcal{C}_1) = \bigcap_{C \in \mathcal{C}_1} V/C$.

Definition 2.2.5. *The intersection of all $\mathcal{D}$-reducts is called the $\mathcal{D}$-core.*

We observe that none of the elements of the core can be removed without affecting the classification power of the attributes.

2.2.3 Partial Dependencies in Knowledge Bases

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

Proposition 2.2.2. $POS_{IND(\mathcal{P})} IND((\mathcal{Q})) = U$ *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 $\mathcal{P} \Rightarrow_k \mathcal{Q}$.

2.3 Standard Dual Relational Generalisation of Rough Set Approximations

In the literature, for example in [94], relational extension of rough sets for an arbitrary binary relation were investigated. In [94], Yao defined such a generalisation as follows: Let R be a binary relation on a set V i.e. $R \subseteq V \times V$. First, the notion of a *successor neighbourhood* of an element $x \in V$, $R_s(x)$ was defined as follows:

$$R_s(x) = \{y \in V \mid xRy\} \tag{2.6}$$

This was then used to define the corresponding notion of lower and upper approximation operators as below. We will use similar notation as in Section 1.2 but note that the subscript here can be any relation not just an equivalence relation. For $X \subseteq V$ we have:

$$\boldsymbol{l}_R(X) = \{x \mid R_s(x) \subseteq X\}$$

$$\boldsymbol{u}_R(X) = \{x \mid R_s(x) \cap X \neq \emptyset\} \tag{2.7}$$

In the Table 2.1, we enlist the properties that different special relations may satisfy. A similar such table was given in [74] and we use it here for comparison with the non-dual relational generalisation examined in the following section. This is a table of properties

	R	R_r	R_s	R_t	R_{rs}	R_{rt}	R_{st}	R_{rst}	R_{ser}
1. Duality of $\boldsymbol{l}_R(X)$, $\boldsymbol{u}_R(X)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
2. $\boldsymbol{l}_R(\emptyset) = \emptyset$	✗	✓	✗	✗	✓	✓	✗	✓	✓
3. $\emptyset = \boldsymbol{u}_R(\emptyset)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
4. $\boldsymbol{l}_R(V) = V$	✓	✓	✓	✓	✓	✓	✓	✓	✓
5. $\boldsymbol{u}_R(V) = V$	✗	✓	✗	✗	✓	✓	✗	✓	✓
6. $\boldsymbol{l}_R(X) \subseteq X$	✗	✓	✗	✗	✓	✓	✗	✓	✗
7. $X \subseteq \boldsymbol{u}_R(X)$	✗	✓	✗	✗	✓	✓	✗	✓	✗
8. $X \subseteq Y \Rightarrow \boldsymbol{l}_R(X) \subseteq \boldsymbol{l}_R(Y)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
9. $X \subseteq Y \Rightarrow \boldsymbol{u}_R(X) \subseteq \boldsymbol{u}_R(Y)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
10. $\boldsymbol{u}_R(X \cup Y) = \boldsymbol{u}_R(X) \cup \boldsymbol{u}_R(Y)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
11. $\boldsymbol{l}_R(X \cap Y) = \boldsymbol{l}_R(X) \cap \boldsymbol{l}_R(Y)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
12. $\boldsymbol{l}_R(X \cup Y) \supseteq \boldsymbol{l}_R(X) \cup \boldsymbol{l}_R(Y)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
13. $\boldsymbol{u}_R(X \cap Y) \supseteq \boldsymbol{u}_R(X) \cap \boldsymbol{u}_R(Y)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
14. $\boldsymbol{l}_R(\boldsymbol{l}_R(X)) \subseteq \boldsymbol{l}_R(X)$	✗	✓	✗	✗	✓	✓	✗	✓	✗
15. $\boldsymbol{l}_R(\boldsymbol{l}_R(X)) \supseteq \boldsymbol{l}_R(X)$	✗	✗	✗	✗	✗	✓	✗	✓	✗
16. $\boldsymbol{u}_R(\boldsymbol{l}_R(X)) \subseteq \boldsymbol{l}_R(X)$	✗	✗	✗	✗	✗	✗	✗	✓	✗
17. $\boldsymbol{u}_R(\boldsymbol{l}_R(X)) \supseteq \boldsymbol{l}_R(X)$	✗	✓	✗	✗	✓	✓	✗	✓	✗
18. $\boldsymbol{u}_R(\boldsymbol{u}_R(X)) \subseteq \boldsymbol{u}_R(X)$	✗	✗	✗	✓	✗	✓	✓	✓	✗
19. $\boldsymbol{u}_R(\boldsymbol{u}_R(X)) \supseteq \boldsymbol{u}_R(X)$	✗	✓	✗	✗	✓	✓	✗	✓	✗
20. $\boldsymbol{l}_R(\boldsymbol{u}_R(X)) \subseteq \boldsymbol{u}_R(X)$	✗	✓	✗	✗	✓	✓	✗	✓	✗
21. $\boldsymbol{u}_R(X) \subseteq \boldsymbol{l}_R(\boldsymbol{u}_R(X))$	✗	✗	✗	✗	✗	✗	✗	✓	✗
22. $X \subseteq \boldsymbol{l}_R(\boldsymbol{u}_R(X))$	✗	✗	✓	✗	✓	✗	✓	✓	✗
23. $\boldsymbol{u}_R(\boldsymbol{l}_R(X)) \subseteq X$	✗	✗	✓	✗	✓	✗	✓	✓	✗

Table 2.1: Properties satisfied by the general approximation operators for different special relations

versus different types of relations. A box is marked with a tick if all relations of the type corresponding to its column satisfies the property stated in its row and is marked with a cross otherwise. Different properties follow for different special relations. Let r, s, t, be subscripts which denote when a relation is reflexive, symmetric and transitive respectively and their combinations denote the conjunction of these properties. Let the subscript ser denote a serial relation i.e. a relation in which every element has a successor.

We note that if a property is satisfied by any general relation, i.e. there is a tick in the first column, then the full row corresponding to that property is ticked. Also, if for example, if some property is satisfied by a reflexive relation i.e. R_r ticked then we can immediately deduce that R_{rs}, R_{rt}, and R_{rst} should be ticked. Similarly, for other special relations. So, often only the first few boxes of a row needs to be figured out before the whole row can be deduced. For example, consider the case of $\boldsymbol{u}_R(\boldsymbol{u}_R(X)) \subseteq \boldsymbol{u}_R(X)$ in the 18th row and the R_t column, i.e for a transitive relation. We now briefly prove this. Suppose that $x \in \boldsymbol{u}_R(\boldsymbol{u}_R(X))$, i.e. $R_s(x) \cap \boldsymbol{u}_R(X) \neq \emptyset$. Let v be in this intersection. Then $v \in R_s(x)$ and $R_s(v) \cap X \neq \emptyset$. So let $t \in R_s(v) \cap X$. Since R is transitive then we also have that $t \in R_s(x)$. Hence, $t \in R_s(x) \cap X$ and $R_s(x) \cap X \neq \emptyset$. Therefore, $x \in \boldsymbol{u}_R(X)$ and $\boldsymbol{u}_R(\boldsymbol{u}_R(X)) \subseteq \boldsymbol{u}_R(X)$. It follows that boxes corresponding to R_t, R_{rt}, R_{st} and R_{rst} are ticked. Counter-example cases can be made for the boxes marked with a cross. We note that not all the rows are independent. For example, $\boldsymbol{l}_R(\boldsymbol{l}_R(X)) \subseteq \boldsymbol{l}_R(X)$ in the 14th row is a special case of $\boldsymbol{l}_R(X) \subseteq X$ in the 6th row. However, we wanted to include both sides of the idempotent equation, consisting of rows 14 and 15, so the complete picture is easier to see. Similar considerations go for the rest of the table.

Examining Table 2.1, we observe a few things. Duality, property 1. as well as properties, 2, 3, 8-13 hold for arbitrary relations. The table also hints at the upcoming 3 propositions. In rows 6, 23 and 18, we see properties which hold for reflexive (but not symmetric and transitive), symmetric (but not reflexive and transitive) and transitive (but not reflexive and symmetric relations) respectively. Forming the table helps to see what possibilities would be promising to try to see if it holds both ways and it can be seen that not only do properties 6, 18. and 23. imply that a relation is reflexive, symmetric and transitive respectively but that the converses hold as well. In the [93] paper, these propositions can be deduced from examinations of algebraic operators. Here, we give brief direct proofs of them and in the next section we will compare these results with what can be obtained for the case of the non-dual generalisation examined.

Proposition 2.3.1. *Let V be a set and R a relation on V. Then $\boldsymbol{l}_R(X) \subseteq X$ for all $X \subseteq V$ iff R is reflexive.*

Proof. $\Leftarrow$ is straightforward so we prove the converse. We prove it by the contrapositive. Suppose that R is not reflexive. Then there exists a witness $x \in V$ such that $(x, x) \notin R$. Consider the set $Y = R_s(x)$. Now by definition $x \in \boldsymbol{l}_R(Y)$ but by assumption $x \notin Y$. Hence $\boldsymbol{l}_R(Y) \not\subseteq Y$. □

Remark 2.3.1 In [105], Zhu also noted the above proposition for characterising approximations for reflexive relations as well as another proposition which characterises reflexive approximations using property 5. in the Table 2.1 instead of property 4.

Proposition 2.3.2. *Let V be a set and R a relation on V. Then $\boldsymbol{u}_R(\boldsymbol{l}_R(X)) \subseteq X$ for all $X \subseteq V$ iff R is symmetric.*

Proof. $\Leftarrow$ is straightforward so we prove the converse. We prove it by the contrapositive. Suppose that R is not symmetric. Then there exits witnesses $x, y \in V$ such that $(x, y) \in R$ but $(y, x) \notin R$. Consider the set $Y = R_s(y)$. By definition we have that $y \in \boldsymbol{l}_R(Y)$. Since $(x, y) \in R$ then $x \in \boldsymbol{u}_R(\boldsymbol{l}_R(Y))$ and since $(y, x) \notin R$ then x is not in Y. Therefore, $\boldsymbol{u}_R(\boldsymbol{l}_R(Y)) \nsubseteq Y$. Hence the result. □

Proposition 2.3.3. *Let V be a set and R a relation on V. Then $\boldsymbol{u}_R(\boldsymbol{u}_R(X)) \subseteq \boldsymbol{u}_R(X)$ for all $X \subseteq V$ iff R is transitive.*

Proof. $\Leftarrow$ is straightforward so we prove the converse. We prove it by the contrapositive. Suppose that R is not transitive. Then there exists witnesses x, y and $z \in V$ such that $(x, y), (y, z) \in R$ but $(x, z) \notin R$. Consider the set $Z = \{z\}$. Then $\boldsymbol{u}_R(Z)$ contains y and hence $\boldsymbol{u}_R(\boldsymbol{u}_R(Z))$ contains x but since $(x, z) \notin R$, x is not in $\boldsymbol{u}_R(Z)$. Hence $\boldsymbol{u}_R(\boldsymbol{u}_R(Z) \nsubseteq \boldsymbol{u}_R(Z)$. The result follows. □

Theorem 2.3.1. *Let R be a relation on a set V. For all $X \subseteq V$, then*

(i) $\boldsymbol{l}_R(X) \subseteq X$,

(ii) $\boldsymbol{u}_R(\boldsymbol{l}_R(X)) \subseteq X$ *and*

(iii) $\boldsymbol{u}_R(\boldsymbol{u}_R(X)) \subseteq \boldsymbol{u}_R(X)$.

all hold iff R is an equivalence relation.

Proof. This is an immediate corollary of Proposition 2.3.1, Proposition 2.3.2 and Proposition 2.3.3. □

Remark 2.3.2 We note that if we replace property (i) in the above theorem with the property 5. from Table 2.1, namely $X \subseteq \boldsymbol{u}_R(X)$, then we get a similar alternative theorem.

Remark 2.3.3 Sometimes even though the given relation is not an equivalence relation, we may form an induced equivalence relation from the relation itself. The choice of such an equivalence relation is not unique but a natural choice was mentioned in [94] which relates elements which are indistinguishable under the given relation. That is, for a relation $R \subseteq V \times V$ and $A \subseteq V$, we define:

$$x \sim y \;\text{ iff } R_s(x) = R_s(y). \tag{2.8}$$

It is easy to see that $\sim$ is an equivalence relation. This is one way to apply the idea of rough approximations for arbitrary relations on a set. We use the relation itself as a source of knowledge (or attribute) about the given set and factor by indistinguishability of the relation to give us our equivalence relation from which subsets of the domain can be compared using the lower and upper approximation operators. In the next chapter, this idea will be used and extended as there we not only consider indistinguishability with respect to the full relation but we also consider different indistinguishability relations relativised to subsets of the nodes i.e. with respect to part of the given relation.

2.4 Non-Dual Relational Generalisation of Rough Set Approximations

Here we examine the properties of a non-dual coupling of lower and upper relational approximations. Analogous to the definition given in equation (2.6), we now give the definition of a *predecessor neighbourhood* of an element $x \in V$, $R_p(x)$, as follows:

$$R_p(x) = \{y \in U|\ yRx\}. \tag{2.9}$$

We will use the lower and upper approximation definitions as follows:

$$\boldsymbol{l}_R(X) = \{x \mid R_s(x) \subseteq X\}$$

$$\mathrm{u}_R(X) = \{x \mid R_p(x) \cap X \neq \emptyset\} \tag{2.10}$$

To emphasize that this is a different upper approximation than the standard generali-

	R	R_r	R_s	R_t	R_{rs}	R_{rt}	R_{st}	R_{rst}	R_{ser}
1. Duality of $\boldsymbol{l}_R(X)$, $\mathrm{u}_R(X)$	✗	✗	✓	✗	✓	✗	✓	✓	✗
2. $\boldsymbol{l}_R(\emptyset) = \emptyset$	✗	✓	✗	✗	✓	✓	✗	✓	✓
3. $\emptyset = \mathrm{u}_R(\emptyset)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
4. $\boldsymbol{l}_R(V) = V$	✓	✓	✓	✓	✓	✓	✓	✓	✓
5. $\mathrm{u}_R(V) = V$	✗	✓	✗	✗	✓	✓	✗	✓	✗
6. $\boldsymbol{l}_R(X) \subseteq X$	✗	✓	✗	✗	✓	✓	✗	✓	✗
7. $X \subseteq \mathrm{u}_R(X)$	✗	✓	✗	✗	✓	✓	✗	✓	✗
8. $X \subseteq Y \Rightarrow \boldsymbol{l}_R(X) \subseteq \boldsymbol{l}_R(Y)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
9. $X \subseteq Y \Rightarrow \mathrm{u}_R(X) \subseteq \mathrm{u}_R(Y)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
10. $\mathrm{u}_R(X \cup Y) = \mathrm{u}_R(X) \cup \mathrm{u}_R(Y)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
11. $\boldsymbol{l}_R(X \cap Y) = \boldsymbol{l}_R(X) \cap \boldsymbol{l}_R(Y)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
12. $\boldsymbol{l}_R(X \cup Y) \supseteq \boldsymbol{l}_R(X) \cup \boldsymbol{l}_R(Y)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
13. $\mathrm{u}_R(X \cap Y) \supseteq \mathrm{u}_R(X) \cap \mathrm{u}_R(Y)$	✓	✓	✓	✓	✓	✓	✓	✓	✓
14. $\boldsymbol{l}_R(\boldsymbol{l}_R(X)) \subseteq \boldsymbol{l}_R(X)$	✗	✓	✗	✗	✓	✓	✗	✓	✗
15. $\boldsymbol{l}_R(\boldsymbol{l}_R(X)) \supseteq \boldsymbol{l}_R(X)$	✗	✗	✗	✗	✗	✓	✗	✓	✗
16. $\mathrm{u}_R(\boldsymbol{l}_R(X)) \subseteq \boldsymbol{l}_R(X)$	✗	✗	✗	✓	✗	✓	✗	✓	✗
17. $\mathrm{u}_R(\boldsymbol{l}_R(X)) \supseteq \boldsymbol{l}_R(X)$	✗	✓	✗	✗	✓	✓	✗	✓	✗
18. $\mathrm{u}_R(\mathrm{u}_R(X)) \subseteq \mathrm{u}_R(X)$	✗	✗	✗	✓	✗	✓	✓	✓	✗
19. $\mathrm{u}_R(\mathrm{u}_R(X)) \supseteq \mathrm{u}_R(X)$	✗	✓	✗	✗	✓	✓	✗	✓	✗
20. $\boldsymbol{l}_R(\mathrm{u}_R(X)) \subseteq \mathrm{u}_R(X)$	✗	✓	✗	✗	✓	✓	✗	✓	✗
21. $\boldsymbol{l}_R(\mathrm{u}_R(X)) \supseteq \mathrm{u}_R(X)$	✗	✗	✗	✓	✗	✓	✓	✓	✗
22. $X \subseteq \boldsymbol{l}_R(\mathrm{u}_R(X))$	✓	✓	✓	✓	✓	✓	✓	✓	✓
23. $\mathrm{u}_R(\boldsymbol{l}_R(X)) \subseteq X$	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 2.2: Properties satisfied by the alternative general approximation operators for different special relations

sation, we use a different font to denote the upper approximation, u_R. In this case, the upper approximation of a set consists of all the successors of elements in that set instead of all the predecessors of elements in that set as in the standard generalisation.

Different properties follow for different special relations. Again, let r, s, t be subscripts which denote when a relation is reflexive, symmetric and transitive and respectively and their combinations denote the conjunction of these properties. Also, let the subscript ser a serial relation.

Consider Table 2.2. Like before, properties 3,4, 8-13 hold for arbitrary relations. However, here we see that duality, property 1. of the table does not hold for arbitrary relations like it does for the standard relational generalisation. On the hand, $X \subseteq \boldsymbol{l}_R(\mathrm{u}_R(X))$ and $\mathrm{u}_R(\boldsymbol{l}_R(X)) \subseteq X$, properties 22. and 23. respectively, does hold for arbitrary relations unlike for the case of the standard relational generalisation. We can also see that the

R_{rst} column, i.e. the column corresponding to an equivalence relation satisfies all of the properties as expected. However, here there is another column of interest which we would like to draw your attention to, namely the column corresponding to R_{rt}. This corresponds to a pre-order and we observe that this satisfies all of the examined rough set properties except the duality of the lower and upper approximation operators. This feature makes it quite interesting and worthy of further consideration.

In Section 2.2, we mentioned characterising properties of the standard relational generalisation which imply R is an equivalence relation. Here, we have characterising properties of the non-dual relational generalisation which imply that R is a pre-order.

Proposition 2.4.1. *Let V be a set and R a relation on V. Then $\mathrm{u}_R(X) \subseteq \boldsymbol{l}_R(\mathrm{u}_R(X))$ for all $X \subseteq V$ iff R is transitive.*

Proof. $\Leftarrow$ is straightforward so we prove the converse. We prove it by the contrapositive. Suppose that R is not transitive. Then there exists witnesses x, y and $z \in V$ such that $(x, y), (y, z) \in R$ but $(x, z) \notin R$. Consider the set $Y = \{x\}$. Then since $(x, y) \in R$, we have that $y \in \mathrm{u}_R(Y)$ and since z is a successor of y but $(x, z) \notin R$ then $y \notin \boldsymbol{l}_R(\mathrm{u}_R(Y))$. Hence, $\mathrm{u}_R(Y) \not\subseteq \boldsymbol{l}_R(\mathrm{u}_R(X))$ and the result follows. □

Theorem 2.4.1. *Let R be a relation on a set V. For all $X \subseteq V$, then*

(i) $\boldsymbol{l}_R(X) \subseteq X$ *and*

(ii) $\mathrm{u}_R(X) \subseteq \boldsymbol{l}_R(\mathrm{u}_R(X))$

both hold iff R is a pre-order.

Proof. This is an immediate corollary of Proposition 2.3.1 and Proposition 2.4.1. □

2.4.1 Applications of the Non-Dual Relational Generalisation

More general context for a special operator satisfying almost all rough set properties

An investigation suggested in [74] asked the question why a certain C_t operator defined in that paper, satisfies so many properties of rough approximation operators based on equivalence relations. Here, amongst other things, they considered covering generalisations of rough sets and they defined the neighbourhood of an element x as all the intersection of cover sets which contain x. That is:

Definition 2.4.1. *Let $\mathcal{C} = \{C_i : i \in I\}$ be a covering of V. Then a neighbourhood of a point $x \in V$ is given by:*

$$N(x) = \bigcap_{i\in I}\{C_i \in \mathcal{C} \mid x \in C_i\}.$$

Now we recall from that paper, special lower and upper approximation operators, in their notation, $\underline{C_t}$ and $\overline{C_t}$, which satisfies all of their mentioned properties of approximation operators based on equivalence relations, except duality. They mentioned that this made this lower and upper approximation pair of operators worthy of further investigation. Here, we show that their operator is a special case of the non-dual relational generalisation which we examined in the previous section. First we recall their defined approximation operators below:

Definition 2.4.2. *Let V be the domain and for $x \in V$, a set $D \subseteq V$ is said to be* definable *if $D = \bigcup_{x\in D} N(x)$. The collection of definable sets is denoted by, $\mathfrak{D} = \{D \subseteq V \mid D$ is definable$\}$. The lower and upper approximation operators, $\underline{C_t}$, $\overline{C_t}$, respectively, given in [74] are as follows:*

$$\begin{aligned}\underline{C_t}(X) &= \bigcup\{D \in \mathfrak{D} \mid D \subseteq X\}\\ &= \bigcup\{N(x) \mid N(x) \subseteq X\},\end{aligned}$$

$$\begin{aligned}\overline{C_t}(X) &= \bigcap\{D \subseteq \mathfrak{D} \mid X \subseteq D\}\\ &= \bigcup\{N(x) \mid x \in X\}.\end{aligned} \tag{2.11}$$

Next, we can see that definitions (2.11) of the lower and upper approximation operators is of the same form as (2.10) of the non-dual generalisation if we take $R_s(x) = N(x)$. Observe that,

$$\begin{aligned}\boldsymbol{l}_R(X) &= \{x \mid R_s(x) \subseteq X\} \\ &= \bigcup\{R_s(x) \mid R_s(x) \subseteq X\},\end{aligned}$$

$$\begin{aligned}\mathrm{u}_R(X) &= \{x \mid R_p(x) \cap X \neq \emptyset\} \\ &= \bigcup\{R_s(x) \mid x \in X\}.\end{aligned} \tag{2.12}$$

We can observe that definition (2.11) looks similar to definition (2.10) with $N(x)$ taking the place of $R_s(x)$. However, in general $R_s(x)$ cannot be considered a neighbourhood of x since we can show from Definition 2.4.1 that $N(x)$ seen as a relation on V is reflexive and transitive, i.e. a pre-order. When we consider the case of R being a pre-order, i.e. R_{rt}, then we can set $R_s(x) = N(x)$ as $N(x)$ is a special case of a pre-order. Hence, we can use Table 2.2 to see which properties hold. From the table we see that all of the usual rough set operator properties except duality holds for R_{rt}. This accounts for the observation of Samanta and Chakraborty in [74] that the operators, $\underline{C_t}$ and $\overline{C_t}$ satisfy all of the rough set properties examined except duality. The reason is that, using the form of approximation operators given in Equation (2.10), any pre-order would satisfy at least those properties satisfied by R_{rt} in Table 2.2.

Interpreted Logical Connection

Consider the case of the pre-order relation being an implication relation on a set of propositions P say. Then, the lower approximation of a subset of $P_1 \subseteq P$ say, corresponds to the union of maximal theories contained in P_1, while the upper approximation corresponds to the smallest theory which contains P_1 i.e. its deductive closure.

Chapter 3

Relativised Indistinguishability Relations using Graphs

In this chapter we will study indistinguishability from many different perspectives. We will mention several indistinguishability notions. Although each of these notions are technically different because of the context in which they are used, they are nonetheless related (behind each of them is essentially the idea in Leibniz's Identity of Indiscernibles, see [30]). In [22], Chiaselotti et al used rough set concepts for approximations in simple, undirected graphs. We briefly go over their setup and results before extending it to the case of undirected graphs with loops. In this case, we may view a graph as simply a relation. Here, the edges can be seen as a relation given on a set of nodes. Indistinguishability by the relation is then defined to give an equivalence relation which is used to compare subsets of the nodes by upper and lower approximations. The setup in [22] goes further than this. They consider not only indistinguishability for the full relation but indistinguishability relativised to any subset of the nodes, i.e. with respect to subsets of the given relation.

Here, we compare the indistinguishability of vertices in a graph relativised to different subsets. We will consider the equivalence classes of single nodes with respect to the different relativised indistinguishability relations. Then, we examine the extremes of possibilities between the equivalence classes of two nodes such that they are always unequal for every relativised indistinguishability relation, with the exception of the equivalence relation consisting of one class i.e. with respect to every non-empty subset of the nodes. We see that this is possible only if the nodes are related to each other in an exactly complementary way. That is, their neighbours partition the vertex set. Also, while it is not possible for three nodes to be always pairwise distinguishable in this way, we will see a construction under which three nodes are pairwise distinguishable for almost all of the relativised relations. We will give the sharp bound for which this is possible.

Next, we list some relativised indistinguishability relation possibilities for hypergraphs and show an application via an interpretation for a database suggesting that the hypergraph setting may be useful for empirical database modelling. Lastly, we notice an equivalence of Cantor's Diagonal Theorem which is connected to indistinguishability and give a discussion of the effects that indistinguishability and vagueness may have on models.

3.1 Graph-theoretic terminology

Definition 3.1.1. *A* graph with possible loops G *consists of a set of nodes or vertices* $V(G)$ *and a set of edges* $E(G)$ *which is binary relation on* $V(G)$. *It can be denoted as* $G = (V(G), E(G))$.

Definition 3.1.2. *A* subgraph $H = (V(H), E(H))$ *of a graph* $G = (V(G), E(G))$ *is a graph such that* $V(H) \subseteq V(G)$ *and* $E(H)$ *is a relation on* $V(H)$ *such that* $E(H) \subseteq E(G)$.

Definition 3.1.3. *Let* $G = (V(G), E(G))$ *be a graph. Then, the* complement *graph of* G *is the graph on the same nodes as* G *but an edge* e *is in the complement graph of* G *iff it is not in* G. *We denote this by* $\overline{G}$. *That is, let* $Fl = V(G) \times V(G)$ *be the full cartesian product on* $V(G)$ *i.e.,* Fl *consists of all pairs of nodes of* G. *Then,* $\overline{G} = (V(G), Fl - E(G))$. *Notice that loopless nodes in a graph will have loops in the graph's complement and vice versa.*

We draw your attention to the difference between the notation for this and for set complement. Let A *be a set. The set-theoretic complement of* A *is denoted by* A'.

Definition 3.1.4. *A* simple graph *is one where identity pairs, i.e. where the co-ordinates are the same node, are not allowed. Simple graphs are also called loopless graphs.*

Definition 3.1.5. K_i *is the simple graph on* i *nodes which consists of each pair of distinct nodes as an edge. These graphs are known as* cliques *or* complete graphs*.*

Note that in the following section we will often use cliques *with* all the loops included.

Definition 3.1.6. *A subset of the vertices of a graph is known as an* independent *set of that graph iff there are no edges which consist of any two vertices of that set.*

Definition 3.1.7. *A* bipartite graph *is a graph whose vertex set can be partitioned into two independent sets.*

Definition 3.1.8. *A* n–partite graph *is a graph whose vertex set can be partitioned into* n *independent sets.*

Definition 3.1.9. *A* complete bipartite graph *is a n-partite graph which consists of all possible edges between the two independent sets of vertices. If the size of the independent sets are* m *and* n*, then this graph is denoted by* $K_{m,n}$*.*

Definition 3.1.10. *A* complete n–partite graph *is a bipartite graph which consists of all possible edges between each pair of the independent sets. If the size of the independent sets are* $n_1, n_2, \dots n_m$ *then this graph is denoted by* $K_{n_1,n_2 \dots n_m}$*.*

Definition 3.1.11. *A* path *in a graph* $V(G)$ *between two nodes* x *and* y *consists of a sequence of edges* $(x, e_1), (e_1, e_2) \dots (e_n, y) \in E(G)$ *such that* $e_i \neq e_j$ *for* $i \neq j$*.*

Definition 3.1.12. *A graph is* connected *if there is a path between any two vertices.*

Definition 3.1.13. *A* component *of a graph is a maximally connected subgraph.*

Definition 3.1.14. *Let* G *be a graph and let* $x \in V(G)$*. Then* $N_G(x)$ *is the* successor neighbourhood *of* x *defined by:*

$$N_G(x) = \{y| \; xRy\} = \{y| \; (x, y) \in R\}. \tag{3.1}$$

When it is clear which graph we are referring to we may leave out the subscript.

3.2 Approximating by Relativised Indistinguishability Relations

In [22], Chiaselotti et al defined an equivalence relation , I_A^G, taken with respect to a subset $A \subseteq V(G)$, induced on undirected loopless graphs. Here, we extended it to consider graphs possibly with loops. For $v,\ w \in V(G)$, we have $(v,w) \in I_A^G$ iff $N(v) \cap A = N(w) \cap A$. That is, the neighbourhoods of v and w are A-indistinguishable. We also denote for convenience I_A^G as the set of the equivalence classes. The corresponding lower and upper approximation operators are:

$$\boldsymbol{l}_{I_A^G}(Y) = \{v \in V(G) \mid [v]_{I_A^G} \subseteq Y\},$$
$$\boldsymbol{u}_{I_A^G}(Y) = \{v \in V(G) \mid [v]_{I_A^G} \cap Y \neq \emptyset\}. \tag{3.2}$$

When it is clear which graph we are referring to, we will write $\boldsymbol{l}_{I_A^G}$ and $\boldsymbol{u}_{I_A^G}$ simply as $\boldsymbol{l}_A$ and $\boldsymbol{u}_A$. We note then that the above are equivalent to the following:

$$\boldsymbol{l}_A(Y) = \{v \in V(G) \mid \forall u \in V(G), (N_G(u) \cap A = N_G(v) \cap A) \Rightarrow u \in Y\},$$
$$\boldsymbol{u}_A(Y) = \{v \in V(G) \mid \exists u \in Y : N_G(u) \cap A = N_G(v) \cap A\}. \tag{3.3}$$

The A-lower approximation of a set $Y \subseteq V(G)$, contains vertices for which all A-indistinguishable vertices from them, i.e. vertices with the same A-neighbourhood, are contained in Y. The upper approximation of Y is the set of vertices each of which share at least one A-connection with a vertex in Y. From the above characterisation they calculated two results on the special cases of cliques (without loops) and complete graphs. We repeat them below.

Proposition 3.2.1. *Let $G = K_n$ be the simple complete graph on n vertices. Let A and Y be two non-empty subsets of $V(G)$. Then,*

(i) the A-lower approximation of Y is,

$$\boldsymbol{l}_A(Y) = \begin{cases} Y & \textit{if } A' \subseteq Y \\ Y \cap A & \textit{otherwise.} \end{cases}$$

(ii) the A-upper approximation of Y is,

$$\boldsymbol{u}_A(Y) = \begin{cases} Y & \textit{if } Y \subseteq A \\ Y \cup A' & \textit{otherwise.} \end{cases}$$

(iii) Y is A-exact if and only if $Y \subseteq A$ *or* $Y \subseteq A'$.

Proposition 3.2.2. *Let* $K_{p,q} = (B_1|B_2)$ *be a bipartite graph. Let A and Y be two non-empty subsets of* $V = V(K_{p,q})$ *such that* $Y \neq V$. *Then, the lower and upper approximation operators are independent of A and are as follows:*

(i) the A-lower approximation of Y is,

$$\boldsymbol{l}_A(Y) = \begin{cases} B_1 & \textit{if } B_1 \subseteq Y \\ B_2 & \textit{if } B_2 \subseteq Y \\ \emptyset & \textit{otherwise.} \end{cases}$$

(ii) the A-upper approximation of Y is,

$$\boldsymbol{u}_A(Y) = \begin{cases} B_1 & \textit{if } Y \subseteq B_1 \\ B_2 & \textit{if } Y \subseteq B_2 \\ V & \textit{otherwise.} \end{cases}$$

(iii) Y is A-exact if and only if $Y = B_1$ *or* $Y = B_2$.

Next, we generalise the above result to the case of n-partite graphs.

Proposition 3.2.3. *Let* $K_{p_1,p_2,\ldots p_n} = (B_1|B_2|,\ldots B_n)$ *be the complete n-partite graph such that* $B_i = \{x_1, x_2, \ldots x_{p_i}\}$ *where* $p_i \geq 1$ *for* $i = 1, 2, \ldots n$. *Let A and Y be two non-empty subsets of* $V = V(K_{p_1,p_2,\ldots p_n})$. *Then,*

(i) the A-lower approximation of Y is,

$$\boldsymbol{l}_A(Y) = \bigcup_i \{B_i : \; B_i \subseteq Y\}$$

(ii) the A-upper approximation of Y is,

$$\boldsymbol{u}_A(Y) = \bigcup_i \{B_i : \; B_i \cap Y \neq \emptyset\}$$

(iii) Y is A-exact if and only if $Y = \bigcup_{i \in I} B_i$ *for some* I*, where* $I \subseteq \{1, 2, \ldots n\}$.

Proof. Straightforward generalisation of the proof given in [22]. □

We now consider the result for cliques *with* loops. This graph can be seen as more symmetrical than a clique without loops and hence the simpler form of the following result as compared with Proposition 3.2.1.

Proposition 3.2.4. *Let* $G = K_n$ *be the complete graph on* n *vertices* with *loops at each node. Let* A *and* Y *be two non-empty subsets of* $V(G)$*. Then,*

(i) the A-lower approximation of Y is,

$$\boldsymbol{l}_A(Y) = \begin{cases} \emptyset & if\ Y \neq K_n \\ K_n & otherwise. \end{cases}$$

(ii) the A-upper approximation of Y is,

$$\boldsymbol{u}_A(Y) = K_n$$

(iii) Y is A-exact if and only if $Y = K_n$.

Proof. Straightforward. □

Remark 3.2.1: The result in Proposition 3.2.4 looks like the result for the special case $n = 1$ in Proposition 3.2.3. The reason is because in the $n = 1$ case for Proposition 3.2.3 we have a 1-partite graph which is a set of independent nodes. For a set of p independent nodes, the complementary graph is the clique with loops on p nodes. Though not the same, it can be seen that these relations result in the same distinguishing of nodes, i.e. they each do not distinguish between any two nodes and therefore equivalence classes induced by these relations with respect to a given subset are the same. Hence, they induce the same lower and upper approximations. This brings us to the following observation.

Proposition 3.2.5. *Let G_1 and G_2 be two graphs. Then,*

$$I_A^{G_1} = I_A^{G_2} \nRightarrow G_1 = G_2.$$

Proof. Using the example of a p independent graph and K_p as discussed above, we see that they generate the same lower and upper approximations though they are not equal. $\square$

Prompted by the previous remark and example, we now show that in general complementary graphs generate the same lower and upper approximations.

Proposition 3.2.6. *Let G and $\overline{G}$ be complement graphs. Then for any fixed $A \subseteq V(G)$, $I_A^G = I_A^{\overline{G}}$.*

Proof. Let $A \subseteq V(G)$, such that $A = \{a_1, a_2 \dots a_n\}$. Suppose that x and y are in the same equivalence class in I_A^G. Then $N_G(x) \cap A = N_G(y) \cap A$. For $\overline{G}$, note that $N_{\overline{G}}(x) = V(G) - N_G(x)$ and $N_{\overline{G}}(y) = V(G) - N_G(y)$.

Let $z \in N_{\overline{G}}(x) \cap A$. $\Rightarrow z \in N_{\overline{G}}(x)$ and $z \in A \Rightarrow z \notin N_G(x)$ and $z \in A \Rightarrow z \notin (N_G(x) \cap A) \Rightarrow z \notin (N_G(y) \cap A)$ by assumption $\Rightarrow z \notin N_G(y)$ since $z \in A \Rightarrow z \in N_{\overline{G}}(y)$ $\Rightarrow z \in N_{\overline{G}}(y) \cap A$ since $z \in A$. Hence x and y are in the same equivalence class in $I_A^{\overline{G}}$ and $I_A^G \leq I_A^{\overline{G}}$. Similarly for the converse i.e. $I_A^{\overline{G}} \leq I_A^G$ and we have that $I_A^G = I_A^{\overline{G}}$. $\square$

Proposition 3.2.7. *If given I_A^G for all $A \subseteq V(G)$, based on an unknown G, then two graph solutions for it can reconstructed, where the graphs are complements of each other.*

Proof. The singleton sets almost determine the whole edge relation because each node splits the graph into two equivalence classes in the equivalence relation relativised to the singleton set consisting of that node. The two equivalence classes partition the nodes of the graph into those which are connected to that node and those which are not. The problem is that we do not know which equivalence class corresponds to connections and which to non-connections. However, we will see that if we consider a fixed node and assume that one particular equivalence class represents the connections to it, then for all other equivalence relations based on singletons, the connection class is determined.

Let $v \in V$ and $A_v = \{v\}$. Suppose that x and y are related in $I_{A_v}^G$. Then either (i) both x any y are connected to v or (ii) neither x nor y is connected to v. In the first case, we will call the equivalence class of x and y the connection class and in the second case we will call it the non-connection class. Assume that (i) is the case. Then for any $w \in V$, with respect to $I^G_{A_v}$, either w is in the connection class or it is not. Let $A_w = \{w\}$, then if w is connected to v, then connection class in $I^G_{A_w}$ is the class which contains v while if w is not connected to v then the non-connection equivalence class in I^G_A is the class which contains v. In this way, assuming the connection or non-connection of an equivalence class of an equivalence relation relativised to a singleton node determines the connection class of all the other equivalence classes based on singleton nodes. Since this information is the neighbourhoods of the nodes, after our choice of (i), the graph is determined. It is easy to see, if we assume option (ii) instead of (i) that the graph obtained will be the complement of the graph determined in (i) since whether an equivalence class with respect a singleton subset, is a connection or non-connection set here, is exactly the reverse of what it would be if (i) were the case. □

The following is essentially a restatement of the above proposition.

Corollary 3.2.1. *Let G and H be (undirected)graphs on a vertex set V. If $I^G_A = I^H_A$ for all $A \subseteq V(G)$, then either $G = H$ or $G = \overline{H}$.*

Corollary 3.2.1 and Proposition 3.2.6 results in the following equivalence.

Theorem 3.2.1. *Let G and H be (undirected) graphs on a vertex set V. Then, $I^G_A = I^H_A$ for all $A \subseteq V(G)$ iff either $G = H$ or $G = \overline{H}$.*

Remark 3.2.2 Prompted by this theorem, we may ask about the characterising prospects with respect to one $A \subseteq V(G)$, instead with respect to having all such subsets. The following propositions will show that the number of graphs on a vertex set with equivalent I^G_A for a fixed $A \subseteq V(G)$ is so large that it indicates the unlikeliness of finding "nice" characterising properties for such graphs.

Proposition 3.2.8. *Let G be a graph with vertex set V and A a fixed subset of G. Then $|I_A| \leq 2^{|A|}$.*

Proof. The equivalence classes of I_A are generated by subsets of A and so they are at most $2^{|A|}$. □

Proposition 3.2.9. *Let V be a set of nodes, $A \subseteq V$ and E a fixed equivalence relation on V such that $|E| \leq 2^{|A|}$. Then there exists (at least one) graph G on V such that $I_A^G = E$.*

Proof. Consider a fixed injection $F : E \to \mathcal{P}(A)$. This exists since $|E| \leq 2^{|A|}$. We form a graph, G from this as follows. To do this we start with the V nodes. If $F([e]) = A_i$, let $A_i = \{a_{i1}, a_{i2}, \dots a_{in}\}$. Then, for any $x \in [e]$ add an edge (x, a_{ij}) for all $j = 0, 1, 2 \dots n$. For this graph, if $F([e]) = A_i$ we have that $x, y \in [e]$ iff $N(x) \cap A = N(y) \cap A = A_i$. The graph formed G, is such that $I_A^G = E$. □

Corollary 3.2.2. *Let V be a set of nodes with $|V| = n$, $A \subseteq V$ such that $|A| = a$ and E a fixed partition on V such that $|E| = m$ and $|E| \leq 2^{|A|}$. Then, there exists $m!\binom{2^a}{m}2^{\binom{n-a}{2}}$ labelled graphs on V such that $I_A^G = E$.*

Proof. We count the number of injections $F : E \to \mathcal{P}(A)$ in the proposition above as each injection gives rise to different A connections in the constructed graph. This is $m!\binom{2^a}{m}$. We can then complete the edges in the graph by forming any other connections not including nodes from A. This gives the $2^{\binom{n-a}{2}}$ factor. □

Remark 3.2.3 Let V be a vertex set with a fixed subset A. Let two graphs, G, H on V be related if $I_A^G = I_A^H$. The above result shows that this is a very coarse relation on the set of graphs on the vertex set V, suggesting that it could be very difficult to find characterising conditions of graphs which are equivalent under this relation. It is however, an open problem to find, if they exist, "nice" graph properties to characterise graphs which are equivalent under this relation. That is:

Open Problem: Consider undirected graphs on a finite vertex set V. Let A be a non-empty subset of V. Find characterising condition of graphs which have equal neighbourhood relations when relativised to A. In other words, if G and H are graphs on V such that $I_A^G = I_A^H$, what is the most that we can say about them?

Next, we will work out restrictions for how many relativised equivalence relations for which two nodes of a graph can be indistinguishable.

Proposition 3.2.10. *Let G be a graph with vertex set V and let $A, B \subseteq V$. If $B \supseteq A$, then $I_B \leq I_A$. That is, I_B is a finer equivalence relation than I_A.*

Proof. Suppose $x, y \in V$ such that $x \sim_{I_B} y$. Then x and y neighbourhoods restricted to B, i.e. their B-neighbourhoods are the same. Hence, their intersection with any subset of B, in particular the subset A, is the same. That is, $x \sim_{I_A} y$. □

The equivalence relation mentioned in (2.8), in the first section is a special case of this.

Corollary 3.2.3. *Let G be a graph with vertex set V and $A \subset V$. Then for $x, y \in V$ the equivalence relation defined by $x \sim y$ iff $N(x) = N(y)$, we have that $\sim \leq \sim_{I_A}$.*

Proof. Apply $A = V$ in Proposition 3.2.10 to obtain the result. □

When $A \subset V$ we say that the nodes of V are classified or distinguished by a *local* context, namely A. When $A = V$ we say that the nodes are classified or distinguished by a *global* context. It is consistent with our expectations that more information gives us finer discrimination.

Proposition 3.2.11. *Let $\sim$ be the equivalence relation obtained from I_A when $A = V$. That is, $x \sim y$ iff $N(x) = N(y)$. Suppose $[x], [y] \in I_V$ such that $[x] \neq [y]$. Then, the equivalence relation I_S induced by $S = (N(x) \cap N(y)) \cup (N(x) \cup N(y))'$ is the finest set-induced equivalence relation that is of the form I_A for which there exists a $[z]_{I_S} \subseteq V$ such that $[z] \supseteq [x]_{I_V} \cup [y]_{I_V}$ is an equivalence class.*

Proof. That an equivalence relation of the form I_A exists for which contains an equivalence class $[z]_{I_A} \supseteq [x]_{I_V} \cup [y]_{I_V}$ is always the case. We simply set $A = \emptyset$. This induced equivalence relation contains one class which is all of V. We show that for any $u, v \in [x]_{I_V} \cup [y]_{I_V}$, $u \sim_{I_S} v$. Suppose that u and v are both in $[x]_{I_V}$ or $[y]_{I_V}$, then they are $\sim_{I_S}$ related by Proposition 3.2.10. Suppose WLOG, $u \in [x]_{I_V}$ and $v \in [y]_{I_V}$. Then $N_S(u) = N_S(v) = S$ by definition of S and $u \sim_{I_S} v$. Hence, I_S contains an equivalence class $[z]_{I_S}$ such that $[z]_{I_S} \supseteq [x]_{I_V} \cup [y]_{I_V}$.

Suppose there is a subset $T \subseteq V$ such that $|T| \geq |S|$. Then there exists a $t \in T$ such that $t \notin (N(x) \cap N(y)) \cup (N(x) \cup N(y))' (= S)$. Therefore, either $t \in N(x)$ but

$t \notin N(y)$ or $t \in N(y)$ but $t \notin N(x)$. In either case, this implies that $N(x) \cap T \neq N(y) \cap T$ and $x \not\sim_{I_T} y$ and thus I_S is the finest such relation in which x and y are in the same equivalence class. □

The proposition below shows us that for a graph G, two vertices whose edge connections are such that they are distinguishable by any subset of $V(G)$, are in fact quite closely related as the edge connections of one is equal to the other in the complement graph.

Proposition 3.2.12. *Let G be a graph. Let $x, y \in V(G)$. Then, $[x] \neq [y]$ for any non-empty $A \subseteq V(G)$ where $[x], [y] \in I_A$ iff $N_G(x) = N_{\overline{G}}(y)$.*

Proof. Suppose that $x, y \in V(G)$ are such that for any $A \subseteq V(G), [x] \neq [y]$ where $[x], [y] \in I_A$. Then in particular for $A = \{a\}$ for any fixed $a \in G$, $(x, a) \in E(G)$ iff $(y, a) \notin E(G)$ iff $(y, a) \in E(\overline{G})$. That is, $N_G(x) = N_{\overline{G}}(y)$.

Conversely, suppose that $N_G(x) = N_{\overline{G}}(y)$. Consider any non-empty $A \subseteq V(G)$. Now, $[x] = [y]$ for $[x], [y] \in I_A$ iff $x, y \in V(G)$ share the same connections with any subset of A i.e are indistinguishable by any subset of A. Since A is non-empty we may consider the equivalence class of I_A induced by $\{a\} \subseteq A$ for some $a \in A$. By the condition, we have that $(x, a) \in E(G)$ iff $(y, a) \notin E(G)$. Hence, edge connections to this subset of A distinguish x and y and thus $[x] \neq [y]$ in I_A. □

Corollary 3.2.4. *Let G be a graph. Let $x, y \in V(G)$. Then, $[x] \neq [y]$ for any non-empty $A \subseteq V(G)$ where $[x], [y] \in I_A$ iff $[x] \neq [y]$ in I_A for each singleton $A \subseteq V$.*

Proof. Immediate from proof of the previous proposition. □

Corollary 3.2.5. *Let G be a graph. Let $x, y, z \in V(G)$. Then it is not possible for each pair of $[x], [y]$ and $[z]$ to be unequal in I_A for each $A \subseteq V(G)$.*

Proof. From the proof in Proposition 3.2.10, we see that if $[x] \neq [y]$ for all $A \subseteq V$ then $N_G(y) = N_{\overline{G}}(x)$. If also, $[z] \neq [x]$ for all $A \subseteq V$ then also $N_G(z) = N_{\overline{G}}(x) = N_G(y)$. Hence $[z] = [y]$ in I_A for all $A \subseteq V$. □

We may call pairs $x, y \in V(G)$ of a graph G *(globally) distinguishable* if $[x] \neq [y]$ in I_V. We say that x and y are *(locally) A–distinguishable* if $[x] \neq [y]$ in I_A where $A \subset V$. Notice that for $x, y \in V$,

locally distinguishable $\Rightarrow$ *globally distinguishable* but,
globally distinguishable $\not\Rightarrow$ *locally distinguishable.*

Proposition 3.2.13. *Let G be a graph and A, a non-empty subset of $V(G)$. Suppose that for some $x, y \in V(G)$ that $[x] \neq [y]$ in I_A. Then, there exists an $a \in A$ such that for the singleton consisting of it, $A_1 = \{a\}$, $[x] \neq [y]$ in I_{A_1}.*

Proof. We prove this by the contrapositive. Suppose that $[x] = [y]$ in I_{A_i} for all $A_i = \{a_i\}$ where $a_i \in A$.

Let a be an arbitrary element in A. Then for $A_1 = \{a\}$, $[x] = [y]$ by the above assumption. That is, $(x, a) \in E(G)$ iff $(y, a) \in E(G)$. Hence, $N_G(x) \cap A = N_G(y) \cap A$. Therefore $[x] = [y]$ in I_A. □

Corollary 3.2.6. *Let G be a graph and A, a non-empty subset of $V(G)$. Then if for some $x, y \in V(G)$, $[x] \neq [y]$ in I_A, then there exists a subset of $\mathcal{B}$ of $\mathcal{P}(V(G))$ such that $[x] \neq [y]$ in I_{B_1} where B_1 is any element of $\mathcal{B}$ and $|\mathcal{B}| \geq \frac{1}{2}|\mathcal{P}(V(G))|$.*

Proof. By Proposition 3.2.13, if $[x] \neq [y]$ in I_A, then there exists an $a \in A$ such that $[x] \neq [y]$ in I_{A_1} where $A_1 = \{a\}$. It is clear that any subset of B_1 of $V(G)$ which contains a will be such that $[x] \neq [y]$ in I_{B_1}. The number of such subsets is $2^{|V(G)|-1}$ which is $\frac{1}{2}|\mathcal{P}(V(G))|$. Hence, the set of subsets of $\mathcal{P}(G)$, B, such that $[x] \neq [y]$ in I_{B_i} where $B_I \in B$, is at least as large as this. □

Corollary 3.2.7. *Let G be a graph and let $\mathcal{B} \subseteq \mathcal{P}(V(G))$ such that $|\mathcal{B}| > \frac{1}{2}|\mathcal{P}(V(G))|$. If for some $x, y \in V(G)$, $[x] = [y]$ in I_{B_i}, for all $B_i \in \mathcal{B}$, then $[x] = [y]$ in I_A for all $A \subseteq V(G)$.*

Proof. Follows immediately from the above corollary. □

Remark 3.2.4 Consider a graph G on at least two nodes. By the above corollary it is not possible that for some $x, y \in V(G)$, that $[x] \neq [y]$ in I_A for only one $A \subseteq V(G)$. If they are unequal in an equivalence relation generated by some subset of $V(G)$, then they are unequal in at least half of the equivalence relations generated in this way. The same is not true for equality. It is possible for $[x] = [y]$ in I_A for exactly one subset A of $V(G)$. To form such a graph, we take a special node $a \in V(G)$ and for the chosen x and y nodes, connect exactly one of $\{x, y\}$ to a. For all remaining nodes we connect them to x and y in the same way. That is, for $v \in V(G)$, such that $v \neq a$, $(x, v) \in E(G)$ iff $(y, v) \in E(G)$. In this way we can form examples of graphs which contain a pair of vertices which is in the same equivalence class for exactly one of these relativised equivalence relations.

Remark 3.2.5 From Corollary 3.2.5. we, see that it is not possible for a graph to contain 3 nodes, each pair of which is distinguishable in all the equivalence relations generated by subsets of the vertices. Clearly, we can find one subset based equivalence relation such that each pair of some three vertices are related (for example let G be a graph with $V = \{a, b, c, d, e\}$ and $E = \{(c, b), (d, a), (e, a), (e, b)\}$ and let $A = \{a, b\}$). Then each pair of $[c], [d], [e] \in I_A$ are unequal). Hence the question becomes what is the maximum size of a subset B of $\mathcal{P}(V(G))$ such that each pair of some three vertices are unequal in each $B_i \in B$? The next definition helps to frame that question concisely.

Definition 3.2.1. *Let **NE(n, k)** equal the maximum size of a subset B of $\mathcal{P}(V(G))$ such that it is possible that for some k unequal vertices of a graph G with |V(G)| = n, to be such that each pair of corresponding equivalence classes in I_{B_I} for each $B_i \in B$, are unequal. For example we know that, $NE(n, 2) = 2^n - 1$ since by Proposition 3.2.12 in this construction inequality holds relative to all subsets of the domain except for the empty set.*

Also let $\boldsymbol{NE}(G, k)$ be the maximum size of a subset B of $\mathcal{P}(V(G))$ such that it is possible that for some k unequal vertices of G nodes, to be such that each pair of equivalence classes in I_{B_I} for each $B_i \in B$, are unequal.

Proposition 3.2.14. *Let G be graph on n nodes where $n \geq 3$. $NE(n, 3) \geq 2^{n-2}$*

Proof. We consider a graph construction as follows. Let x, y and z be the three nodes who will witness $NE(n, 3)$. Let $a_1, a_2 \in V(G)$ be two special nodes. We form connections between the three nodes and the two special nodes as follows. Let, $(x, a_1), (y, a_2) \in E(G)$ and let there be no other edges between the three nodes and the two special ones. Then for $A = \{a_1, a_2\}$, $N_G(x) \cap A = \{a_1\}$, $N_G(y) \cap A = \{a_2\}$ and $N_G(z) \cap A = \emptyset$. Hence each pair of $[x], [y]$ and $[z]$ are unequal in I_A. Now, any $A_1 \supseteq A$ will be such that each pair of $[x], [y]$ and $[z]$ is unequal in I_{A_1}. There are 2^{n-2} subsets of $\mathcal{P}(V(G))$ that contain these two nodes and hence $NE(n, 3) \geq 2^{n-2}$. □

While Corollary 3.2.5. shows that all subsets cannot simultaneously distinguish between three nodes, the following theorem shows it is possible for almost all subsets to do so.

Theorem 3.2.2. *Let G be graph on n nodes where $n \geq 3$. Then $\boldsymbol{NE}(n, 3) \leq 2^n - 3.2^{\frac{n}{3}} + 2$ with equality holding when n is divisible by three.*

Proof. We construct a maximal graph with three special nodes x, y and z which witness the result. We begin with an observation. For each singleton node $a \in V(G)$, there are 4 possibilities. Let $A = \{a\}$. Then in I_A either 1) $[x] = [y] = [z]$, 2) $[x] = [y]$, $[x] \neq [z]$ and $[y] \neq [z]$, 3) $[x] = [z]$, $[x] \neq [y]$ and $[z] \neq [y]$ or 4) $[y] = [z]$, $[y] \neq [x]$ and $[z] \neq [x]$.

Observation: Notice that any two nodes which satisfies any pair of the last three properties distinguishes pairwise between x, y and z.

Claim: In a maximal construction, i.e maximal for $\boldsymbol{NE}(n, 3)$, there are no nodes say, a, such that for $A = \{a\}$, $[x] = [y] = [z]$ in I_A. Suppose there was such a node a in a maximal graph G on n nodes. Let $\boldsymbol{NE}(n, 3) = m$ which G realizes. Since $[x] = [y] = [z]$ in I_A where $A = \{a\}$, then either all three of x, y and z are connected to a or all three are unconnected to a. We form a new graph G^1 from G on the same n nodes by either removing a connection from x to a in the first case or adding a connection from x to a in the second case. For both of these cases we have now $[y] = [z]$, $[x] \neq [y]$ and $[x] \neq [z]$ in I_A for G^1. Hence $\{a\}$ now distinguishes pairwise between x and either of y, z.

Now, consider a node $b \in G^1$ such that either 2) $[x] = [y]$, $[x] \neq [z]$ or $[y] \neq [z]$, 3) $[x] = [z]$, $[x] \neq [y]$. A node satisfying one of these must exist since a maximal graph contains at least one subset which distinguishes between x, y and z (that such graphs exists is clear–check the proof of Proposition 3.2.14) and any subset which distinguishes between these contains a minimal subset which does so and this is a pair as in the Observation above. Since this pair satisfies two of 2), 3) 4) listed above, then one of them satisfies one of 2) or 3). Now it is clear to see that {a, b} which does not pairwise distinguish between x, y and z in G does so in G^1. Since all of the subsets of $\mathcal{P}(V(G))$ which previously distinguished between each pair of x, y and z in G still do in G^1 we get that $\boldsymbol{NE}(G^1, 3) > \boldsymbol{NE}(G, 3)$ with $|V(G')| = |V(G)|$ contradicting maximality of G and the claim is shown.

From the above Claim, we see that each node in a maximal construction, G, is one that satisfies either 2), 3) or 4). So we can partition the nodes of the graph into 3 sets corresponding to nodes which satisfy 2), 3) and 4) respectively. From the observation above, any two nodes from distinct parts of the partition distinguishes between x, y and z. So to count the subsets of $V(G)$ which distinguishes between these three nodes we note that any subset besides the emptyset and the ones totally contained in one part of the partition minus the emptyset to avoid overcounting it, will do. Let the sizes of the partition be s, t and $n - (s + t)$. We observe again the partition must at least have two parts to distinguish between all three nodes. This graph can be formed for example by letting s nodes be joined to exactly z in $\{x, y, z\}$ to satisfy 2), t nodes be joined to exactly y in $\{x, y, z\}$ to satisfy 3) and $n - (s + t)$ nodes be joined to exactly x in $\{x, y, z\}$ to satisfy 4). Here the number of subsets of $\mathcal{P}(V(G))$ which distinguish the three nodes

are therefore,

$$\begin{aligned}\boldsymbol{NE}(G, 3) &= 2^n - 1 - ((2^s - 1) + (2^t - 1) + (2^{n-(s+t)} - 1)) \\ &= 2^n - (2^s + 2^t + 2^{n-(s+t)}) + 2.\end{aligned}$$

Hence to maximise $\boldsymbol{NE}(G, 3)$ we need to minimise $(2^s + 2^t + 2^{n-(s+t)})$. Since 2^m grows very rapidly with m this amounts to minimising the maximum of $\{s, t, n - (s + t)\}$. It is clear that to do this we have to take as equal parts as possible and they would be of size $\frac{n}{3}$ when n is divisible by three. Here, $\mathbf{NE}(n, 3) = 2^n - 3.2^{\frac{n}{3}} + 2$. Otherwise the value of

$\mathbf{NE}(n,3)$ is slightly less than this when $n \equiv 1$ or $n \equiv 2$ **mod** 3. □

Now we can give a slight strengthening of Corollary 3.2.5

Corollary 3.2.8. *Let G be a graph n nodes and let $\mathcal{B} \subseteq \mathcal{P}(V(G))$ such that $|\mathcal{B}| > 2^n - 3.2^{\frac{n}{3}} + 2$. Suppose that and x, y, and z are three nodes in $V(G)$. Then, there exists a $B_i \in \mathcal{B}$ such that at least two of x, y and z, are equivalent in I_{B_i}.*

3.2.1 Extensions of Relativised Indistinguishability Relations in Hypergraphs

Here, we extend the setup introduced in [22] to the hypergraph setting. For recent work in the literature on hypergraphs and rough sets, see [23], where hypergraphs are used to connect rough sets to granular computing and [20] where connections between rough sets and formal concept analysis are made amongst other results. In our case, we extend the set-up in [22] to consider several other natural options for which equivalence relations can be based on when using hypergraphs. We list some examples of these and give a database interpretation of this setting which can be used for modelling. Depending on one's modelling aim, one can choose the equivalence relation best suited to define the lower and upper approximations on. First we give some definitions.

Definition 3.2.2. *A* hypergraph *is a pair, $\mathcal{H} = (V, \mathcal{E})$, where $\mathcal{V}$ is a set and $\mathcal{E}$ is a collection of subsets of $\mathcal{V}$. If the same subset of V cannot be selected more than once, the $\mathcal{E}$ is a set and the hypergraph is said to be a* simple hypergraph.

Definition 3.2.3. *The* degree *of a vertex $v \in V$, denoted by $d(v)$, is the number of edges which contain v.*

Definition 3.2.4. *A* regular hypergraph *is one for which, $d(x) = d(y)$ for all $x, y \in V$.*

Definition 3.2.5. *A* t-regular hypergraph *is a regular hypergraph with common degree t.*

Definition 3.2.6. *A* uniform hypergraph *is one for which $|e| = |f|$ for all $e, f \in \mathcal{E}$.*

Definition 3.2.7. *A* t-uniform hypergraph *is a uniform hypergraph with edges of* t *vertices.*

Definition 3.2.8. *We observe that* simple graphs *can be defined as 2-uniform hypergraphs without loops.*

We define different relativised equivalence relations similar to that given in [22] for the case of hypergraphs. Here, there are more possibilities of equivalence classes on nodes to consider and we shall mention and compare several of these below. Now, we give the definitions:

Definition 3.2.9. *Let* $\mathcal{H} = (V, \mathcal{E})$ *be a hypergraph. The* vertex neighbourhood *of a vertex* x *in a hypergraph is as follows:*

$$N^v(x) = \{z \in V \mid \exists\ e \in \mathcal{E} : \{x, z\} \subseteq e\}. \tag{3.4}$$

This definition can be extended to give a neighbourhood of $A \subseteq V$ in a natural way as follows:

$$N^v(A) = \bigcup_{x \in A} N^v(x). \tag{3.5}$$

Definition 3.2.10. *Let* $\mathcal{H} = (V, \mathcal{E})$ *be a hypergraph. The* edge neighbourhood *of a vertex* x *in a hypergraph is as follows:*

$$N^e(x) = \{e \in \mathcal{E} \mid x \in e\}. \tag{3.6}$$

In a similar manner to the preceding, this definition can be extended to give a neighbourhood of $A \subseteq V$ in a natural way as follows:

$$N^e(A) = \bigcup_{x \in A} N^e(x). \tag{3.7}$$

From the above definitions it can be seen that we have the information of $N^v(x)$ from $N^e(x)$. It is clear that,

$$N^v(x) = \bigcup_{e \in N^e(x)} e. \tag{3.8}$$

However, $N^e(x)$ cannot be recovered from $N^v(x)$.

Next we consider several ways which we can associate vertices relative to a subset A of the vertices.

Let G be a graph and $A \subseteq \mathcal{V}$ and let $x \in \mathcal{V}$. Then we define certain A-neighbourhoods of x as follows:

1. $N_A^v(x) = N^v(x) \cap A$
2. $N_A^e(x) = N^e(x) \cap N^e(A) = \{e \in \mathcal{E} \mid (x \in e) \wedge (e \cap A \neq \emptyset)\}$
3. $N_A^t(x) = \{e \cap A \mid (x \in e) \wedge (e \cap A \neq \emptyset)\}$
4. $N_A^1(x) = \{a \in A \mid \exists\ e \in \mathcal{E} : (x \in e) \wedge (e \cap A = \{a\})\}$
5. $N_A^k(x) = \{a_1 \in A \mid \exists a_2, \dots a_k \in A,\ \exists\ e \in \mathcal{E} : (x \in e) \wedge (e \cap A = \{a_1, a_2 ... a_k\})\}$
6. $N_A^{lk}(x) = \{a_1 \in A \mid \exists a_2, \dots a_k \in A\ for\ j \leq k,\ \exists\ e \in \mathcal{E} : (x \in e) \wedge (e \cap A = \{a_1, a_2 ... a_j\})\}$
7. $N_A^{gk}(x) = \{a_1 \in A \mid \exists a_2, \dots a_k \in A\ for\ j \geq k,\ \exists\ e \in \mathcal{E} : (x \in e) \wedge (e \cap A = \{a_1, a_2 ... a_j\})\}$

We observe that for *graphs*, $N_A^1 = N_A^v$ which is equivalent to the neighbourhood definition used in the previous section. The respective equivalence relations based on the above definitions are as follows:

1.$(x, y) \in I_A^v$ iff $N_A^v(x) = N_A^v(y)$

2.$(x, y) \in I_A^e$ iff $N_A^e(x) = N_A^e(y)$

3.$(x, y) \in I_A^t$ iff $N_A^t(x) = N_A^t(y)$

4.$(x, y) \in I_A^1$ iff $N_A^1(x) = N_A^1(y)$

5.$(x, y) \in I_A^k$ iff $N_A^k(x) = N_A^k(y)$

6.$(x, y) \in I_A^{lk}$ iff $N_A^{lk}(x) = N_A^{lk}(y)$

7.$(x, y) \in I_A^{gk}$ iff $N_A^{gk}(x) = N_A^{gk}(y)$

We now compare the fineness of these equivalence relations induced on the vertices.

Proposition 3.2.15. *Let $\mathcal{H} = (V, \mathcal{E})$ be a hypergraph and $A \subseteq V$. Then, I_A^e implies all the other defined relations. That is; (i) $I_A^e \leq I_A^v$, (ii) $I_A^e \leq I_A^1$, (iii) $I_A^e \leq I_A^t$, (iv)$I_A^e \leq I_A^k$, (v) $I_A^e \leq I^l k_A$, and (vi) $I_A^e \leq I^g k_A$.*

Proof. For (i) : Suppose that $(x, y) \in I_A^1$. Then $N_A^e(x) = N_A^e(y)$. Equation (23) shows that $N_A^v(x)$ can be recovered from $N_A^e(x)$. Hence, $N_A^v(x) = N_A^v(y)$ holds and $(x, y) \in I_A^v$.

For (ii): Suppose that $(x, y) \in I_A^e$. Then $N_A^e(x) = N_A^e(y)$. That is, each edge which contains x also contains y. Now if $a \in N_A^1(x)$, then there exists an edge $e \in \mathcal{E}$ such that e contains x and $e \cap A = \{a\}$. By assumption e contains y as well and so $a \in N_A^1(y)$. Thus, $N_A^1(x) \subseteq N_A^1(y)$. By symmetry, the converse holds as well and hence $N_A^1(x) = N_A^1(y)$ which gives $(x, y) \in N_A^1$.

Part (iii) follows immediately and parts (iv)-(vi) can be proved similarly to part (ii). □

It is interesting that apart from the very fine relation I_A^1, the other comparisons between the relations show them to be quite independent. For example, even though $I_A^v = I_A^1$ holds in the case of *graphs*, no implication holds between either side in the case of general hypergraphs. Also, even though one might initially expect that $I_A^{lk} \leq I_A^k$, this too does not hold (of course the converse does not hold). We briefly illustrate these observations with counterexamples.

Proposition 3.2.16. *(i)* $I_A^v \not\leq I_A^1$ *and (ii)* $I_A^1 \not\leq I_A^v$.

Proof. Consider a hypergraph, $\mathcal{H} = (V, \mathcal{E})$ such that $V = \{a, b, x, y, z\}$, $\mathcal{E} = \{\{x, a\}, \{x, b\}, \{y, a, b\}\}$ and $A = \{a, b\}$. Then, $N_A^v(x) = N_A^v(y) = \{a, b\}$. But $N_A^1(x) \neq N_A^1(y)$ since $N_A^1(x) = \{a, b\}$ while $N_A^1(y) = \emptyset$. Hence, $I_A^v \not\leq I_A^1$.

Also, $N_A^1(y) = N_A^1(z) = \emptyset$. But $N_A^v(y) \neq N_A^v(z)$ since $N_A^v(y) = \{a, b\}$ while $N_A^v(z) = \emptyset$. Hence, $N_A^1 \not\leq N_A^v$. □

Proposition 3.2.17. $I_A^{lk} \not\leq I_A^k$.

Proof. Consider a hypergraph, $\mathcal{H} = (V, \mathcal{E})$ such that $V = \{a, b, c, d, x, y\}$ and $\mathcal{E} = \{\{x, a, b\}, \{x, c\}, \{x, d\}, \{y, b, c\}, \{y, a, d\}\}$. Let $A = \{a, b, c, d\}$ and $k = 2$. Then, $N_A^{l2}(x) = N_A^{l2}(y) = \{a, b, c, d\}$. But $N_A^2(x) \neq N_A^2(y)$ since $N_A^2(x) = \{a, b\}$ and $N_A^2(y) = \{a, b, c, d\}$. Hence, $I_A^{lk} \not\leq I_A^k$. □

That no implication holds between all other of the relation pairs defined other than those given in Proposition 3.2.15 (i.e between I_A^e and any of the other ones) can be shown with similar counterexamples to the two preceding propositions. The above considerations indicate that I_A^e is one of the finest natural ways to identify the vertices of a hypergraph. The other equivalence relations considered are independent ways to associate the vertices and we may choose the one most suited to our needs/model in a given situation.

Proposition 3.2.18. *Let $\mathcal{H} = (V, \mathcal{E})$ be a hypergraph. Let $x, y \in V$. Suppose that $[x] = [y]$ in $I_{A_x}^e$ and in $I_{A_y}^e$ where $A_x = \{x\}$ and $A_y = \{y\}$. Then $[x] = [y]$ in I_V^e.*

Proof. Since $x, y \in V$ and $[x] = [y]$ in $I_{A_x}^e$, where $A_x = \{x\}$, then if an edge contains x then it contains y. Also, since $x, y \in V$ and $[x] = [y]$ in $I_{A_y}^e$ where $A_y = \{y\}$, then if an edge contains y then it contains x. Hence we have that an edge contains x iff it contains y iff $x \sim_{I_V^e} y$. Therefore, $[x] = [y]$ in I_V^e. □

Proposition 3.2.19. *Let $\mathcal{H} = (V, \mathcal{E})$ be a hypergraph. Suppose that for any $x, y \in V$, $[x] \neq [y]$ in I_V^e. Then there exists a separating edge e such that e contains exactly one of $\{x, y\}$. Such hypergraphs are called* separating hypergraphs*.*

Proof. Immediate from definition. □

Proposition 3.2.20. *Let $\mathcal{H} = (V, \mathcal{E})$ be a hypergraph. If there exists $x, y \in V$ such that for any $A \subseteq V$, $[x] \neq [y]$, where $[x], [y] \in I_A^t$, then,*

(i) x and y are two vertices not connected by any edge,

(ii) for any $e \in \mathcal{E}$, e contains exactly one of $\{x, y\}$

(iii) for any $v \in V$, $\{v, x\}$ is contained in at least one edge and $\{v, y\}$ is not contained in any edge, or $\{v, y\}$ is contained in at least one edge and $\{v, x\}$ is not contained in any edge. That is, x and y partitions V into vertices which are connected to exactly one of them.

Proof. Since $[x] \neq [y]$, where $[x], [y] \in I_A^t$, for any $A \subseteq V$, then consider the particular case of $A = \{x\}$. The only subsets of A are $\{x\}$ and $\emptyset$, so since $[x] \neq [y]$ then the N_A^t neighbourhood of one of these nodes is $\{x\}$ while for the other it is $\emptyset$. Suppose that $N_A^t(y) = \{\{x\}\}$. Then there exists an $e \in \mathcal{E}$ such that $e \supset \{x, y\}$. Using this same e we can deduce that $N_A^t(x) = \{\{x\}\} = N_A^t(y)$ which is a contradiction. So we get that $N_A^t(y) = \emptyset$ and $N_A^t(x) = \{\{x\}\}$. That is, there is no edge which contains both x and y

and there is at least one edge which contains x. By a similar argument using $A = y$ we have that there is at least one edge which contains y.

Now suppose that $f \in \mathcal{E}$. Then it cannot be disjoint from both x and y because if it were then let A_1 be the set of vertices contained in f. Then, $N^t_{A_1}(x) = N^t_{A_1}(y) = \emptyset$ which is a contradiction. Also, we know that it can't contain both x any y from the argument above, hence it must contain exactly one of them. That is, x and y partitions the edges of $\mathcal{E}$ into two sets, namely those which contain x but not y and those which contain y but not x.

Suppose that $v \in V$ is such there exists $e \in \mathcal{E}$ which contains x, y and v or there is no edge in $\mathcal{E}$ which contains v. In the first case if we take $A = \{v\}$, then $N^t_A(x) = N^t_A(y) = \{\{v\}\}$ in I^t_A and in the second case $N^t_A(x) = N^t_A(y) = \emptyset$ both of which is a contradiction. Hence, each $v \in V$ is connected by at least one edge to exactly one of $\{x, y\}$. □

Application by Database Interpretation: Empirical Model Theory

We interpret a database by first separating it into sets of Objects and Attributes. Here, we consider an interpretation in which the attributes are nodes and the objects are edges that consists of exactly the attributes which they satisfy. In this case, we consider binary valued attributes, i.e. for a set of attributes, *Att* and a set of objects *Obj*, we are equipped with a binary function $v : Obj \times Att \to \{0, 1\}$. An object $x \in Obj$ has an attribute, $a \in Att$ iff $v(x, a) = 1$ while x does not have attribute a iff $v(x, a) = 0$. From this we can form a corresponding hypergraph from this database, $\mathcal{H} = (Att, Obj)$. That is, the attributes *Att* are the vertices and the objects *Obj* are the edges such that each object contains all the attributes which it satisfies. We now give definitions which can be used for empirical modelling of attributes of a database.

If two attributes are realised by at least one object they are said to be **compatible**. In the hypergraph setting which we outlined, this means that two vertices are compatible if there is at least one edge which contains them.

Two attributes are said to be **incompatible** with respect to a domain of objects if no object realises both of them, i.e no edge from the domain of objects contains those two vertices. This concept is relative to the domain of objects because if the domain of objects is extended, then two previously determined incompatible objects may become compatible under satisfaction from a new object in the extended domain.

Two attributes are said to be **contingent** if there exists two objects one of which contains both of them and one of which contains exactly one of them. That is, two attributes are contingent if they may be satisfied together but not necessarily so.

An attribute a **implies** an attribute b iff every object which satisfies attribute a satisfies attribute b. That is, iff every edge which contains a also contains b. Here we may say $a \Rightarrow b$. If also $b \Rightarrow a$, then $a \iff b$ and we say that attributes a and b are **indistinguishable** by the object domain. Otherwise we say that the attributes are **distinguishable** by the object domain.

Remark 3.2.6: We note the compatibility, contingency and distinguishability of two attributes are preserved under extensions of the domain of objects but incompatibility, dependency and indistinguishability may change.

Example of an Interpreted Result

Under this interpretation, we translate Proposition 3.2.20 as follows:

Proposition 3.2.21. *Let $\mathcal{H} = (Att, Obj)$ be a simple hypergraph and consider the I_A^t equivalence relation. If there exists $a, b \in Att$ which for any $A \subseteq Att$ are never equivalent under this relation then,*

(i) a and b are incompatible attributes,

(ii) Any object $x \in Obj$, satisfies exactly one of $\{a, b\}$,

(iii) For any other attribute $v \in Att$, v is compatible with either a or b but not both.

Similar translations can be made for other results of this setup.

Remark 3.2.7 Instead of examining attribute distinguishability we may alternatively consider object distinguishability. We can consider the graph setup used in this chapter in a special case of Kripke semantics for modal logic. Then, we can define a relation on the set of objects which models relativised distinguishability in the semantics. For basic concepts and definitions of modal logic and Kripke semantics, see [14].

For this case, we consider transforming a model first given as a matrix of *Objects* versus *Attributes* where *Objects* and *Attributes* are finite sets. Let $O \in Objects$ and $A \in Attributes$, then the valuation, $v(A, O) = 1$ iff object O satisfies attribute A and the entry is assigned 0 otherwise (i.e. we consider here two-valued attributes). To translate this information into a Kripke frame, we interpret the propositional variables as the objects, use elements of the powerset of *Attributes*, $\mathcal{P}(Attributes)$ as the worlds and for the accessibility relation R, $(\mathcal{A}_1, \mathcal{A}_2) \in R$ iff $\mathcal{A}_1 \subseteq \mathcal{A}_2$. Now we can define a valuation or a particular model, $\langle W, R, v\rangle$ recursively as follows:

1. If $\mathcal{A} \in \mathcal{P}(Attributes)$, then $\mathcal{A} \models O$ iff there exists an $A \in \mathcal{A}$, $A \models O$ (i.e. $v(A, O) = 1$).
2. $\mathcal{A} \models \neg O$ iff $\mathcal{A} \not\models O$.
3. $\mathcal{A} \models (O_1 \wedge O_2)$ iff $\mathcal{A} \models O_1$ and $\mathcal{A} \models O_2$.
4. $\mathcal{A} \models \Box O$ iff for all $\mathcal{B} \in \mathcal{P}(Attributes)$, if $(\mathcal{A}, \mathcal{B}) \in R$ then $\mathcal{B} \models O$.
5. $\mathcal{A} \models \Diamond O$ iff there exists $\mathcal{B} \in \mathcal{P}(Attributes)$ such that $(\mathcal{A}, \mathcal{B}) \in R$ and $\mathcal{B} \models O$.

We say that two objects O_1 and O_2 are $\mathcal{A}$–*distinguishable* iff there exists an $A \in \mathcal{A}$ such that either $v(A, O_1) = 1$ (equivalently, $A \models O_1$) and $v(A, O_2) = 0$ or $v(A, O_2) = 1$ and $v(A, O_1) = 0$. Otherwise, O_1 and O_2 are said to be $\mathcal{A}-$ *indistinguishable*. We observe that if $\mathcal{A} \subseteq \mathcal{B}$ then,

- O_1 and O_2 are $\mathcal{A}$–distinguishable $\Rightarrow$ O_1 and O_2 are $\mathcal{B}$–distinguishable
- O_1 and O_2 are $\mathcal{B}$–indistinguishable $\Rightarrow$ O_1 and O_2 are $\mathcal{A}$–indistinguishable

That is, relativised distinguishability is upwards absolute and relativised indistinguishability is downwards absolute.

3.2.2 Indistinguishability on Real Numbers

Next, we found a very interesting equivalence to Cantor's Diagonal Theorem which is related to the topic of indistinguishability. We note that Pawlak did some work on rough approximations of real numbers by partitioning the real numbers into intervals in [59].

First we give some definitions:

Definition 3.2.11. *A relation $h : A \to B$ is* well-defined *if for any elements $(a, b), (c, d) \in h, a = c$ implies that $b = d$.*

Definition 3.2.12. *: A relation $h : A \to B$ is* injective *if for any elements $(a, b), (c, d) \in h, b = d$ implies that $a = b$.*

Definition 3.2.13. *A relation $h : A \to B$ is said to be* domain-exhausted *if for any $a \in A$ there exists b in B such that $(a, b) \in h$.*

Definition 3.2.14. *: A relation $h : A \to B$ is said to be* surjective *if for any $b \in B$ there exists a in A such that $(a, b) \in h$.*

Definition 3.2.15. *If h is a relation such that $h : A \to B$, then the* inverse relation*, is h^{-1}, where $h^{-1} : B \to A$ is such that $(b, a) \in h^{-1}$ iff $(a, b) \in h$.*

Definition 3.2.16. *Let h be a relation such that, $h : A \to B$. Then for $a \in A$, $h(a) = \{b \in B \mid (a, b) \in h\}$.*

Definition 3.2.17. *A* function *from sets A to B is a relation from A to B which is domain-exhausted and well-defined.*

Definition 3.2.18. *A* bijection *is an injective and surjective function.*

Definition 3.2.19. *The* domain *of a relation $h : A \to B$, is the set consisting of all $a \in A$ such that there exists $b \in B$ such that $(a, b) \in h$.*

Definition 3.2.20. *The* range *of a relation $h : A \to B$, is the set consisting of all $b \in B$ such that there exists $a \in A$ such that $(a, b) \in h$.*

Let $\mathbb{N}$ denote natural numbers and $\mathbb{R}$ denote real numbers. Next, we state Cantor's Diagonal Theorem.

Theorem 3.2.3. *(Cantor's Diagonal Theorem)*
There is is no bijective function, $f : \mathbb{N} \to \mathbb{R}$.

Next, we show a nice result which Cantor's Diagonal Theorem implies.

Theorem 3.2.4. *If a relation $h : \mathbb{N} \to \mathbb{R}$ is surjective then it is not well-defined.*

Proof. Suppose that there exists a relation $h : \mathbb{N} \to \mathbb{R}$ which is surjective and well-defined. Since the relation is surjective and $|\mathbb{R}|$ is infinite, this means that the domain of h in $\mathbb{N}$ is an infinite set since each of its elements is mapped to exactly one element of $\mathbb{R}$ by well-definedness of h. Let the domain of h in $\mathbb{N}$ be N'. Hence $|N'| = |\mathbb{N}| = \omega$.

Since h is surjective, then for any $r \in \mathbb{R}$, $h^{-1}(r) \neq \emptyset$. Since the relation is well-defined, the inverse sets are disjoint i.e. $h^{-1}(r) \cap h^{-1}(s) = \emptyset$ for $r \neq s$. Then for each $r \in \mathbb{R}$, we can choose a unique element in $h^{-1}(r)$ by choosing least element in that set (we can do this since $h^{-1}(r) \subseteq \mathbb{N}$). Call it h_r. Let N'' be the subset of N' which is such that $N'' = \{n \in N' | \; \exists r \in \mathbb{R} : h_r = n\}$. Since each element of $\mathbb{R}$ is associated with a unique h_r, then $N'' \subseteq \mathbb{N}$ is infinite and the relation $h_R : \mathbb{R} \to N''$ defined by $h_R(r) = h_r$ is a bijection. Hence $|N''| = \omega$ and there exists a bijection $g : \mathbb{N} \to N''$. Also $h_R^{-1} : N'' \to \mathbb{R}$ is a bijection. Thus we can compose relations g followed by h_R^{-1} to get a bijection from $\mathbb{N}$ to $\mathbb{R}$. That is, $(h_R^{-1})(g) : \mathbb{N} \to \mathbb{R}$ is a bijection which is a contradiction to Cantor's Diagonal Theorem. Hence the result is shown. □

Corollary 3.2.9. *Cantor's Diagonal Theorem $\Leftrightarrow$ Theorem 3.2.4.*

Proof. Since the proof of Theorem 3.2.4 needs Cantor's Diagonal Theorem, the "$\Rightarrow$" direction is shown. The converse is immediate since from the theorem we have that there is no well-defined, surjective relation from $\mathbb{N} \to \mathbb{R}$ and so in particular, there is no bijective function from $\mathbb{N} \to \mathbb{R}$. □

Remark 3.2.8 Notice that Theorem 3.2.4 shows that not only can there be no bijective function from $\mathbb{N}$ to $\mathbb{R}$ but there cannot even be any *surjective* function from $\mathbb{N}$ to $\mathbb{R}$ since a function is well-defined.

Using the theorem, we can say that any surjective relation h from $\mathbb{N}$ to $\mathbb{R}$ induces an indistinguishability relation on $\mathbb{R}$ with two elements $r, s \in \mathbb{R}$ being *h-indistinguishable* iff $h^{-1}(r) \cap h^{-1}(s) \neq \emptyset$. Since h is not well-defined there exists some $r, s \in \mathbb{R}$ where $r \neq s$ which is such that $h^{-1}(r) \cap h^{-1}(s) \neq \emptyset$. Hence, this relation is not the identity one. Also, it is easy to check that this is a similarity relation (it is not an equivalence relation in general since the relation need not be injective). Therefore, we have that any surjective relation from $\mathbb{N}$ to $\mathbb{R}$ induces a non-trivial indistinguishability relation on $\mathbb{R}$ and the elements of $\mathbb{N}$ cannot be used to distinguish all the elements of $\mathbb{R}$.

3.2.3 Discussion of Vagueness in Models

"Some people are always critical of vague statements. I tend rather to be critical of precise statements; they are the only ones which can correctly be labeled 'wrong'."

–Raymond Smullyan

The following is a brief discussion which is of the form of an essay rather than of precise definitions and results (which is the format of this thesis with the exception of this section). We apologise that our wish to discuss connections with rough sets and vagueness must also be somewhat vague. However, we think that the ideas that will be outlined are sufficiently interesting and nicely related to the topic to warrant their inclusion in the thesis.

Above we showed that in some sense, $\mathbb{R}$ has an intrinsic indistinguishability with respect to $\mathbb{N}$. We would like to imagine a thought experiment. Suppose that there are two collections. In the first collection A, we can access all of its individual elements while in the second collection B, we can only access equivalence classes on B with respect to some equivalence relation E on B. What if we had no way to access or uniquely name the elements of B or rather what if the names that we have for B are not separated by an identity relation but only a non-trivial indistinguishability relation (notice that when we usually consider an equivalence relation a set, we are also implicitly assuming an equivalence relation on the set at least as fine as to separate the elements of that set). That is, some elements of B would unfortunately have to share a name. Then if a name should appear more than once in a context we could never be sure if it is referring to one object or more than one object. In fact, if the only way we could ever refer to these objects is by possibly referring to any element indistinguishable from it, then it might be a matter of faith than these objects exist in an individuated(/discrete/precise) sense at all. Moreover, if we want to make a model which uses B, we might be, at least implicitly, be inserting vagueness into our model. If we use the names in B then these names can be seen as a clique of possibilities. A non-classical logic might be better suited to describe the behaviour of the model and under the assumption of extra conditions, different extensions or precisifications may be forced. This may remind one of supervaluationism semantics and in fact the analogy with supervaluationism and set-theoretical forcing has been observed by Toby Meadows in [55].

Another point is that the interpretation of the quantifiers, ‘there exists’ and ‘for all’ may need to be modified for the case of modelling this type of behaviour. Here, ‘there exists’ may need to be amended to include the possibility that we might not (sometimes in principle) be able to use or refer to a specific realisation of what exists. For example in the Sorites paradox or the paradox of baldness, the issue is what is the exact number of hairs which is the cut-off point between baldness and non-baldness. We have the seemingly reasonable assumption that someone who has n hairs is bald implies that someone who has $n+1$ hairs is bald. From this we obtain that, to say any point is the cut-off point, is to say the same of its neighbour. There is an underlying indistinguishability structure based on a similarity relation between the hairs with respect to baldness. Another fun example by the prominent logician Raymond Smullyan given in [77] is as follows: Imagine that we are all immortal but there is a disease which if caught, puts one in a deep sleep forever. There is also an antidote which if given n days after one has caught it will allow one to be awake for 2^n days before returning to a deep sleep forever. Suppose now that your love has gotten the disease and that you have the antidote. On which day should you awaken him/her? It is obviously true that you should awaken your partner on *some* day but on any day if you just wait a little more, then you can spend twice the time with your love. So in a sense, there exists *a* day that you should use the antidote however there doesn’t exist a *unique, non-arbitrary* day that you should do so. This is related to the concept of omega-incompleteness in logic. Note also, that with respect to names, it may be possible that we may not even have a unique representation or approximation by which to refer to elements– that is, even the existence of a unique representative for elements may only "weakly" exist as in the manner outlined above. What we are trying to say with these illustrations is that when a structure has a inherent vague/relatively-undefinable/continuous form and we attempt to treat it as if it has a precise/sharp/discernible/discrete form, then we are likely to get a mismatch of syntax and semantics. Hence, we may lose or sweep under the carpet some non-negligible behaviour of the model.

We want to draw your attention to the fact that from the time we define a semantics, the semantics *itself* brings with it implicit assumptions. For example, names and constants, though not explicitly stated so, are usually assumed to be discrete, mutually distinguishable labels for elements of a structure. But what happens if we assume, somewhat carelessly, that we can assign names which have an identity relation on them to elements of a model which intrinsically have a coarser, non-trivial indistinguishability structure. The names would unknowingly make us feel that we have more distinguishability power than we actually have and this mismatch of labels having extra properties than the labell*ed*, could lead to unexpected or surprising results (especially when we can only refer to the labelled by using the labels).

We are hinting that real numbers may have such a structure with respect to assigning it names from $\mathbb{N}$ (or any other structure with isomorphic distinguishability structure to $\mathbb{N}$). So perhaps it is time that we more consciously try to capture the behaviour of these collections by using a tool which embraces non-trivial indistinguishability relations. That tool, could very well be Rough Set Theory.

Chapter 4

Successive Approximations

Successive approximations in this chapter 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. For example, say 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_2L_1. Could we then reconstruct E_1 and E_2 from this information? This is analogous to Fourier Analysis, where decompose waves into simpler sine and cosine waves. In this chapter, we will investigate this question and consider the four cases of being given a defined $L_2L_1,\ U_2U_1,\ U_2L_1,\ L_2U_1$ 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. For the L_2L_1 case we will show that these conditions form a preclusive relation between pairs of equivalence relations on a set and so we can define a related notion of independence from it. Also, in section 4.2.3 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.

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 4.0.1. *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. $\boldsymbol{u}_{E_1}(\boldsymbol{l}_{E_2}(Y)) = Z \quad \not\Rightarrow \quad \boldsymbol{l}_{E_2}(\boldsymbol{u}_{E_1}(Y)) = Z$,

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 $\boldsymbol{l}_{E_2}(\boldsymbol{u}_{E_1}(Y)) = \{a, b\}$.

□

4.1 Properties of Successive Approximations

We list some further properties of successive approximations below.

From Properties 1), 5) and 6) of lower and upper approximations in Section 2.2.1 in chapter 2, 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 $E_1 \leq E_2$ then the successive approximations are constrained as follows:

Proposition 4.1.1. *If $E_1 \leq E_2$ 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. $\square$

Proposition 4.1.2. *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 4.1.1. *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 4.1.2. *Let V be a finite non-empty set and let E_1 and E_2 be equivalence relations on V. Then $POS_{E_1}(E_2) = \bigcup_{x \in V} \boldsymbol{l}_{E_1}(\boldsymbol{u}_{E_2}(\{x\}))$.*

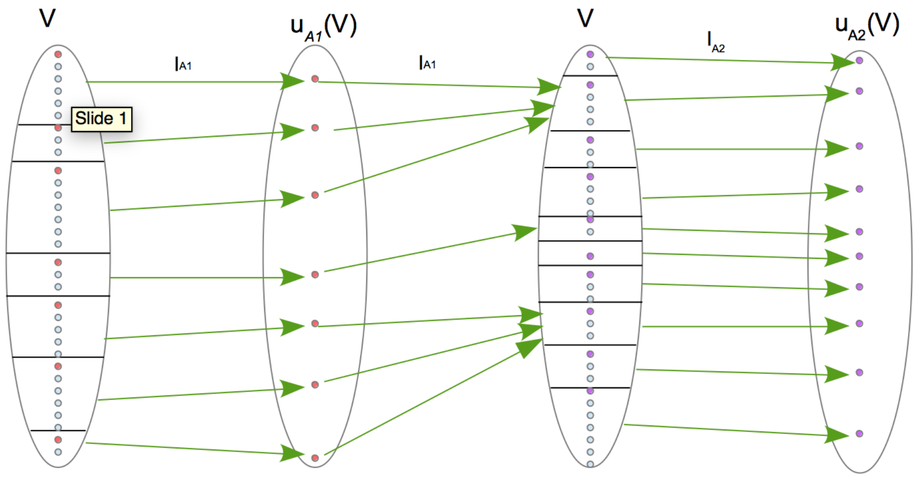

Figure 4.1: Illustrates that successive approximations get more coarse when iterated.

Proposition 4.1.3. *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 $X \subseteq F(X)$. $\square$

Remark 4.1.1. 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.

4.2 Decomposing L_2L_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 4.2.1

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:

$L_2L_1(\{\emptyset\}) = \emptyset$

$L_2L_1(\{a\}) = \emptyset$

$L_2L_1(\{b\}) = \emptyset$

$L_2L_1(\{c\}) = \emptyset$

$L_2L_1(\{d\}) = \emptyset$

$L_2L_1(\{e\}) = \emptyset$

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

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

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

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

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

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

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

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

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

$L_2L_1(\{d, e\}) = \{e\}$

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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 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 4.2.2

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 4.2.1. *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 $L_2L_1([x]) = L_2(\emptyset) = \emptyset$. □

Remark 4.2.1 We note that the union of E-undefinable sets is not necessarily E-undefinable. Consider Example 4.2.2. 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 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 range of L_2L_1 i.e. 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 4.2.2. *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 $Y_t \supseteq Y$. □

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

Lemma 4.2.1. *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 $E_1 \leq S$. $\square$

Remark 4.2.3 The above shows that for any other solution, (E_1, E_2) of a given L_2L_1 operator other than (S, R) produced by the algorithm, must be finer than (S, R), i.e. $E_1 \leq S$ and $E_2 \leq R$.

Lemma 4.2.2. *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 4.2.3. *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). □

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 4.2.1. *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 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* $\boldsymbol{u}_{E_1}([x]_{E_2}) \neq \boldsymbol{u}_{E_1}([y]_{E_2})$*,*

(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) *is the solution 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 4.2.2 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 4.2.3 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 4.2.2 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 4.2.1, 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 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 set in the range 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 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 $L_2L_1(\boldsymbol{u}_{E_1}([a]_{E_2})) \subseteq L_2L_1(\boldsymbol{u}_{E_1}([b]_{E_2}))$ and $L_2L_1(\boldsymbol{u}_{E_1}([a]_{E_2})) \supseteq L_2L_1(\boldsymbol{u}_{E_1}([b]_{E_2}))$ which implies that $L_2L_1(\boldsymbol{u}_{E_1}([a]_{E_2})) = L_2L_1(\boldsymbol{u}_{E_1}([b]_{E_2}))$. 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 $a \not\sim_{E_1} b$. 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 $L_2L_1(X)$ 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. □

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 4.2.1. *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 $E = \{([u]_C, [v]_D) \mid \exists\, x \in V: \ x \in [u]_C$ and $x \in [v]_D\}$. We call this the **incidence graph** of the pair (C, D).*

Theorem 4.2.2. *Let V be a finite set and let $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ 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 complete 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. □

Corollary 4.2.1. *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 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. □

The next natural question is, without assuming conditions on the equivalence relations, are there instances when the algorithm produces a unique solution? Example 4.2.1 is an example of a unique decomposition of a given L_2L_1 operator. So this leads naturally to the next question. What conditions result in a unique solution to a given L_2L_1? Can we find characterising features of the pairs of equivalence relations which give a unique L_2L_1 operator?

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 4.2.1. Recall that in Example 4.2.2, 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 4.2.1, 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.

4.2.1 Characterising Unique Solutions

Theorem 4.2.3. *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 4.2.1, 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.

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 4.2.1, $(E_1, E_2) = (S, R)$. 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 $L_2L_1(Y_m) = Y$, P is contained in Y_m iff Q is contained in Y_m iff $[x]_S$ is contained in Y_m. Hence, $L_RL_S(X) = L_RL_{E_1}(X)$ 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). □

The following theorem sums up the results of Theorem 4.2.1 and Theorem 4.2.3.

Theorem 4.2.4. *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 (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, $|[x]_{E_2} \cap [z]_{E_1}| = 1$.

(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 4.2.4: If an equivalence relation pair satisfies the conditions of Theorem 4.2.1, 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 4.2.3, 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 4.2.1 and Theorem 4.2.3, then the L_2L_1 operator produced by it is unique.

Corollary 4.2.2. *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 4.2.3. □

Example 4.2.1 (*revisited*): Consider again, the given output vector of Example 4.2.1. 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 4.2.1. Now we need only to check if this pair satisfies the conditions of Theorem 4.2.3 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 4.2.3 and therefore this is the unique solution of the given operator.

Proposition 4.2.3. *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 4.2.2 and the second violates the second condition of Theorem 4.2.1. Hence the solutions cannot be unique. Similarly, if we suppose that $E_2 \leq E_1$. □

Corollary 4.2.3. *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* $E_1 \not\Rightarrow E_2$

(ii) $k = \gamma(E_2, E_1) = \frac{|POS_{E_2}(E_1)|}{|V|} < 1$ *or* $E_2 \not\Rightarrow E_1$.

Proof. This follows immediately from definitions. □

Proposition 4.2.4. *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 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 4.2.4 and Corollary 4.2.2 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. □

Corollary 4.2.4. *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 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. □

Corollary 4.2.5. *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 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). □

Proposition 4.2.5. *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 4.2.4 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. □

4.2.2 A Derived Preclusive Relation and a Notion of Independence

Let V be a finite set and let $\mathfrak{E}_V$ be the set of all equivalence relations on V. Also, let $\mathfrak{E}^0_V = \mathfrak{E}_V - Id_V$, 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 $\mathfrak{E}^o$, $\not\Rightarrow_{\mathfrak{E}^o}$, as follows:

Let E_1 and E_2 be in $\mathfrak{E}^o$. Let $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ where L_1 and L_2 are lower approximation operators based on E_1 and E_2 respectively. Then,

$$E_1 \not\Rightarrow_{\mathfrak{E}^o} E_2 \text{ iff } L_2L_1 \text{ is a unique approximation operator.}$$

That is, if for no other E_3 and E_4 in $\mathfrak{E}^o$ where at least one of $E_1 \neq E_3$ or $E_2 \neq E_4$ holds, is it the case that the operator $L_2L_1 = L_3L_4$, where L_3 and L_4 are lower approximation operators based on E_3 and E_4 respectively. For more about the connections between rough set approximations and preclusive relations see [17, 24].

Definition 4.2.2. *Let V be a set and $E_1, E_2 \in \mathfrak{E}^0_V$. We say that E_1 is $\mathfrak{E}^0_V$–**independent** of E_2 iff $E_1 \not\Rightarrow_{\mathfrak{E}^0_V} E_2$. Also, if $\neg(E_1 \not\Rightarrow_{\mathfrak{E}^0_V} E_2)$, we simply write $E_1 \Rightarrow_{\mathfrak{E}^0_V} E_2$. Here, we say the E_1 is $\mathfrak{E}^0_V$–**dependent** of E_2 iff $E_1 \Rightarrow_{\mathfrak{E}^0_V} E_2$.*

Proposition 4.2.6. *$\not\Rightarrow_{\mathfrak{E}^0_V}$ is a preclusive relation.*

Proof. We recall that a preclusive relation is one which is irreflexive and symmetric. Let $E \in \mathfrak{E}^o{}_V$. Since $E \neq Id$, then by application of Proposition 4.2.3 (E, E) does not generate a unique L_2L_1 operator and therefore $E \Rightarrow_{\mathfrak{E}^0_V} E$. Hence $\not\Rightarrow_{\mathfrak{E}^0_V}$ is irreflexive.

Now, suppose that $E_1, E_2 \in \mathfrak{E}^o{}_V$ are such that $E_1 \not\Rightarrow_{\mathfrak{E}^0_V} E_2$. Then (E_1, E_2) satisfies the conditions of Theorem 4.2.4. 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. □

Remark 4.2.5: From the previous proposition we can see that dependency relation $\Rightarrow_{\mathfrak{E}_V^0}$ is a similarity relation.

Proposition 4.2.7. *If $E_1 \Rightarrow E_2$ then $E_1 \Rightarrow_{\mathfrak{E}_V^0} E_2$.*

Proof. This follows from Corollary 4.2.3. □

Proposition 4.2.8. *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 4.2.2 we see (E_1, E_2) does not give a corresponding unique L_2L_1 operator, hence $E_1 \Rightarrow_{\mathfrak{E}_V^0} E_2$ but $E_1 \not\Rightarrow E_2$. □

Remark 4.2.6 From Proposition 4.2.7 and Proposition 4.2.8, 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 4.2.5. *Let V be a finite set and E_1 and E_2 equivalence relations on V. Then $E_1 \not\Rightarrow_{\mathfrak{E}_V^0} E_2$ 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 4.2.4.

□

4.2.3 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 4.2.4 for unique solutions.

Proposition 4.2.9. *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 split 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 $\boldsymbol{u}_{E_2}(Y) = \boldsymbol{u}_{E_2}(Z) = \boldsymbol{u}_{E_2}([x]_{E_1})$. □

Using the preceding proposition we obtain an equivalent form of Theorem 4.2.4.

Theorem 4.2.6. *Let* V *be a finite set and* E_1 *and* E_2 *equivalence relations on* V*. Then* (E_1, E_2) *produces a unique* $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ *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* $\boldsymbol{u}_{E_2}(Y) \neq \boldsymbol{u}_{E_2}(Z)$

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**mod** E_1. That is, let V**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 4.2.4

Theorem 4.2.7. *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_2L_1 operator iff the following holds:*

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

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

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

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

4.3 Decomposing U_2U_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 4.3.1. *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 $\mathcal{P}(V)$. Then any solution (E_1, E_2), is also a solution of $L_2L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ operator where $L_2L_1(X) = -U_2U_1(-X)$ for any $X \in \mathcal{P}(V)$. Therefore, the solution (E_1, E_2) for the defined U_2U_1 operator is a unique iff the solution for the corresponding L_2L_1 operator is unique.*

Proof. Recall that $L_2L_1(X) = -U_2U_1(-X)$. Hence, if there exists a solution (E_1, E_2) which corresponds to the given U_2U_1 operator, this solution corresponds to a solution for the L_2L_1 operator which is based on the same (E_1, E_2) by the equation $L_2L_1(X) = -U_2U_1(-X)$. Similarly for the converse. □

Algorithm: 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 $\mathcal{P}(V)$. To solve for a solution, change it to solving for a solution for the corresponding L_2L_1 operator by the equation $L_2L_1(X) = -U_2U_1(-X)$. Then, when we want to know the L_2L_1 output of a set we look at the U_2U_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_2U_1 operator.

4.3.1 Characterising Unique Solutions

Theorem 4.3.1. *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 4.3.1 using Theorem 4.2.4. □

4.4 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 (see Theorem 4.4.1 and Theorem 4.4.2). 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 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 range of U_2L_1 i.e. 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.

Again, we bring your attention to the fact that this algorithm is different from Algorithm 4.1 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).

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

Lemma 4.4.1. *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 $E_1 = S$. □

Lemma 4.4.2. *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). □

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

Theorem 4.4.1. *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) is the solution 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 4.4.1, $C = S$, 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 4.4.2 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 4.4.1, $E_2 \leq R$.

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 4.4.1).

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 $R = E_2$. □

4.4.1 Characterising Unique Solutions

Theorem 4.4.2. *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 4.4.1 if there is a unique solution, it is (S, R) produced by Algorithm 4.2. We note that by Lemma 4.4.1, 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 4.4.1 and Theorem 4.4.2.

Theorem 4.4.3. *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, $|[x]_{E_2} \cap [z]_{E_1}| = 1$.

4.5 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 4.5.1. *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 $U_2 L_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ 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 $L_2U_1 : \mathcal{P}(V) \to \mathcal{P}(V)$ 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_2L_1 operator by the equation $U_2L_1(X) = -L_2U_1(-X)$. Then, when we want to know the U_2L_1 output of a set we look at the L_2U_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_2U_1 operator.

4.5.1 Characterising Unique Solutions

Theorem 4.5.1. *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 4.5.1, Theorem 4.4.1 and Theorem 4.4.2. □

Conclusion and Future Work

In chapter 2 we found a relational generalisation which satisfies almost all of the usual rough set properties except for duality for the pre-order case. This explains why an operator observed in the literature in [74] is so well-behaved– namely because it is a special case of this construction. Additionally, we found a nice interpreted logical connection for this generalisation. This may be a stepping stone in understanding why this extension is so well-behaved. The future work which follows from this is to develop and investigate this connection.

In chapter 3 we examined relativised indistinguishability using graphs, extending the work in [22]. We found a characterisation theorem which describes which graphs have equivalent relativised indistinguishability relations for all such induced relations for graphs on a given vertex set. We then proposed the open problem of finding characterising features of graphs which are equivalent for one (at least) non-trivial relativised indistinguishability relation for graphs under a given vertex set. Then, we briefly extended the set-up to the hypergraph case. Furthermore, we noticed a nice equivalence of Cantor's theorem and discussed how this is related to indistinguishability differences between $\mathbb{N}$ and $\mathbb{R}$.

In chapter 4 we examined the problem of decomposing double, successive, rough approximations into single rough approximations. We found characterising conditions for equivalence pair solutions which produce unique such operators and noticed that pairs of equivalence relations on a set which produces unique operators form a preclusive relation. We can thus derive a related notion of independence from this. Furthermore, we found a conceptual translation for the conditions of the uniqueness theorem in the L_2L_1 case. An area for future work is to continue to find these nice conceptual translations as suggested by Yao in [99]. We note that most of the results in chapters 2 and 3 do not require V to be a finite set however the algorithms in chapter 4 do have that requirement (for the existence of minimum or minimal sets). Hence, it is a very interesting question how this situation looks and what analysis follows for successive rough set approximations based on equivalence relations on an infinite set.

Bibliography

[1] E. A. Abo-Tabl. A comparison of two kinds of definitions of rough approximations based on a similarity relation. *Information Sciences*, 181:2587–2596, 2011.

[2] E. A. Abo-Tabl. Rough sets and topological spaces based on similarity. *International Journal of Machine Learning and Cybernetics*, 4(5):451–458, 2013.

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

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

[5] A. Avron and B. Konikowska. Rough sets and 3-valued logics. *Studia Logica*, 90:60–92, 2008.

[6] M. Banerjee. Rough sets and 3-valued Łukasiewicz logic. *Fundamenta Informaticae*, 31, 1997.

[7] M. Banerjee and M. K. Chakraborty. Rough sets through algebraic logic. *Fundamenta Informaticae*, 28(3-4), 1996.

[8] M. Banerjee and M. K. Chakraborty. Foundations of vagueness: a category-theoretic approach. *Electronic Notes in Theoretical Computer Science*, 82:10–19, 2003.

[9] M. Banerjee and M. K. Chakraborty. Algebras from rough sets. *Cognitive Technologies, Rough-Neural Computing*, pages 157–184, 2004.

[10] M. Banerjee and M. K. Chakraborty. Rough sets: Some foundational issues. *Fundamenta Informaticae*, 127:1–15, 2013.

[11] R. Bello, R. Falcó, and W. Pedrycz, editors. *Granular Computing: At the Junction of Rough Sets and Fuzzy Sets*. Studies in Fuzziness and Soft Computing (Book 224). Springer, 2008.

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

[13] T. Bittner and J.G. Stell. Stratified rough sets and vagueness. *Lecture Notes in Computer Science, Spatial Information Theory. Foundations of Geographic Information Science*, 2825:270–286, 2002.

[14] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*, volume 53 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 2002.

[15] Z. Bonikowski and U. Wybraniec-Skardowska. Rough sets and vague sets. *Lecture Notes in Computer Science*, 4585:122–132, 2007.

[16] G. Cattaneo. An investigation about rough set theory: Some foundational and mathematical aspects. *Fundamenta Informaticae - Advances in Rough Set Theory*, 108(3-4):197–221, 2011.

[17] 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.

[18] G. Cattaneo and D. Ciucci. Algebraic structures for rough sets. *LNCS Transactions on Rough Sets II*, 3135:218–264, 2004.

[19] G. Cattaneo and D Ciucci. Lattices with interior and closure operators and abstract approximation spaces. *LNCS Transactions on Rough Sets X*, 5656:67–116, 2009.

[20] G. Cattaneo, G. Chiaselotti, D. Ciucci, and T. Gentile. On the connection of hypergraph theory with formal concept analysis and rough set theory. *Inf. Sci.*, 330:342–357, 2016.

[21] M. K. Chakraborty and M. Banerjee. Rough logics : a survey with further directions. In E. Orlowska, editor, *Incomplete Information: Rough Set Analysis*, volume 13 of *Studies in Fuzziness and Soft Computing*, pages 579–600. Springer-Verlag, year.

[22] G. Chiaselotti, D. Ciucci, and T. Gentile. Simple undirected graphs as formal contexts. *Formal Concept Analysis*, pages 287–302, 2015.

[23] G. Chiaselotti, D. Ciucci, and T. Gentile. Simple graphs in granular computing. *Inf. Sci.*, 340-341:279–304, 2016.

[24] G. Chiaselotti, D. Ciucci, T. Gentile, and F. Infusino. Preclusivity and simple graphs. *RSFDGrC*, pages 127–137, 2015.

[25] D. Ciucci and D. Dubois. Truth-functionality, rough sets and three-valued logics. *2010 40th IEEE International Symposium on Multiple-Valued Logic*, pages 98–103, 2010.

[26] C. Cornelis, R. Jensen, D. Slezak, and G. Hurtado. Attribute selection with fuzzy decision reducts. *Information Sciences*, 180(2):209–224, 2010.

[27] 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.

[28] I. Düntsch. A logic for rough sets. *Theoretical Computer Science*, 179:427–436, 1997.

[29] T-F. Fan. Rough set analysis of relational structures. *Information Sciences*, 221:230–244, 2013.

[30] P. Forrest. The Identity of Indiscernibles. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, winter 2016 edition, 2016.

[31] A. Grabowski. Relational formal characterization of rough sets. *Formalized Mathematics*, 21(1):55–64, 2013.

[32] J. W. Grzymala-Busse and W. Rzasa. Definability and other properties of approximations for generalized indiscernibility relations. *LNCS Transactions on Rough Sets XI*, 5946:14–39, 2010.

[33] 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.

[34] 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.

[35] 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.

[36] 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.

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

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

[39] 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.

[40] J. Järvinen. Approximations and rough sets based on tolerances. *LNCS Rough Sets and Current Trends in Computing*, pages 182–189, 2001.

[41] 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.

[42] E.F. Lashin, A.M. Kozae, A.A. Abo Khadra, and T. Medhat. Rough set theory for topological spaces. *International Journal of Approximate Reasoning*, 40(1-2):35–43, 2005.

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

[44] X. Li and X. Yuan. The category rsc of i-rough sets. *FSKD '08 Proceedings of the 2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery*, 1:448–452, 2008.

[45] C. Liau. Modal reasoning and rough set theory. *LNCS Artificial Intelligence: Methodology, Systems, and Applications*, 1480:317–330, 2006.

[46] T.Y. Lin. Topological and fuzzy rough sets. In *Decision Support by Experience—Application of the Rough Sets Theory*, pages 287–304. Kluwer Academic Publishers, 1992.

[47] T.Y Lin. Neighborhood systems: a qualitative theory for fuzzy and rough sets. In P.P. Wang, editor, *Advances in Machine Intelligence and Soft Computing*, pages 132–155. Department of Electrical Engineering, Duke University, Durham, North Carolina, 1997.

[48] T.Y. Lin. *Granular computing: Fuzzy logic and rough sets*, chapter Computing with Words in Information/Intelligent Systems, pages 183–200. Physica-Verlag, 1999.

[49] T.Y. Lin and C. Liau. Granular computing and rough sets. *Data Mining and Knowledge Discovery Handbook*, pages 535–561, 2005.

[50] J. Lu, S. Li, X. Yuang, and W. Fu. Categorical properties of m-indiscernibility spaces. *Theoretical Computer Science*, 412(42):5902–5908, 2011.

[51] M. Ma and M. K. Chakraborty. Covering-based rough sets and modal logics. part i. *International Journal of Approximate Reasoning*, 77:55–65, 2016.

[52] 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.

[53] V. W. Marek. Zdzisław Pawlak, databases and rough sets. In Z. Suraj A. Skowron, editor, *Rough Sets and Intelligent Systems,*, page 175–184. Springer, 2013.

[54] W. Marek and H. Rasiowa. Gradual approximating sets by means of equivalence relations. *Bulletin of Polish Academy of Sciences, Mathematics*, 35:233–238, 1987.

[55] T. Meadows. Forcing for philosophical logicians. `https://sites.google.com/site/tobymeadows/papers-talks`, January 2014. Accessed: 2016-10-30.

[56] 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.

[57] P. Pagliani. Rough set theory and logic-algebraic structures. *Incomplete Information: Rough Set Analysis*, pages 109–190, 1998.

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

[59] Z. Pawlak. Rough sets. *International Journal of Computer and Information Sciences*, pages 341–356, 1982.

[60] Z. Pawlak. Rough sets and fuzzy sets. *Fuzzy Sets and Systems*, 17(1):99–102, 1985.

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

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

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

[64] Z. Pawlak and A. Skowron. Rough sets: Some extensions. *Information Sciences*, 177(1):28–40, 2007.

[65] 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.

[66] 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.

[67] L. Polkowski. *Rough Sets: Mathematical Foundations.* Advances in Intelligent and Soft Computing. Physica, 1 edition, 2002.

[68] L. Polkowski and M. Semeniuk-Polkowska. Where rough sets and fuzzy sets meet. *Fundamenta Informaticae*, 142(1-4):269–284, 2015.

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

[70] K. Qin and Z. Pei. On the topological properties of fuzzy rough sets. *Fuzzy Sets and Systems*, 151(3):601–613, 2005.

[71] M. Restrepo, C. Cornelis, and J. Gómez. Duality, conjugacy and adjointness of approximation operators in covering-based rough sets. *International Journal of Approximate Reasoning*, 55(1 Part 4):469–485, 2014.

[72] A. Salama and H. Abu-Donia. Generalizations of rough functions in topological spaces by using pre-open sets. *Journal of Intelligent Learning Systems and Applications*, 4(2):127–134, 2012.

[73] M. Salamó and M. López-Sánchez. Rough set based approaches to feature selection for case-based reasoning classifiers. *Pattern Recognition Letters*, 32(2):280–292, 2011.

[74] P. Samanta and M. K. Chakraborty. Interface of rough set systems and modal logics: A survey. *Transactions on Rough Sets XIX*, pages 114–137, 2015.

[75] A. Skowron. Rough sets and vague concepts. *Fundamenta Informaticae*, 64(1-4):417–431, 2004.

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

[77] R. M. Smullyan. *A Beginner's Guide to Mathematical Logic (Dover Books on Mathematics).* Dover Publications, 2014.

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

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

[80] H. Thiele. Generalizing the explicit concept of rough set on the basis of modal logic. *Advances in Soft Computing, Computational Intelligence in Theory and Practice*, 8:93–120, 2001.

[81] B.K. Tripathy. Multi-granular computing through rough sets. *Advances in Secure Computing, Internet Services, and Applications*, pages 1–34, 2014.

[82] 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.

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

[84] 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.

[85] D. Vakarelov. A modal logic for similarity relations in Pawlak knowledge representation systems. *Fundamenta Informaticae*, 15:61–79, 1991.

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

[87] P. Wasilewski and D. Slezak. *Rough Computing: Theories, Technologies and Applications*, chapter Foundations of Rough Sets from Vagueness Perspective, pages 1–37. IGI Global, 2008.

[88] L. Wei and W. Zhang. Probabilistic rough sets characterized by fuzzy sets. *Probabilistic rough sets characterized by fuzzy sets*, 2639, 2003.

[89] A. Wiweger. On topological rough sets. *Bulletin of the Polish Academy of Sciences – Mathematics*, 37:51–62, 1988.

[90] M. Wygralak. Rough sets and fuzzy sets-some remarks on interrelations. *Fuzzy Sets and Systems*, 29(2):241–243, 1989.

[91] W. Xu, Q. Wang, and X. Zhang. Multi-granulation rough sets based on tolerance relations. *Soft Computing*, 17(7):1241–1252, 2013.

[92] Y. Y. Yao. A comparative study of fuzzy sets and rough sets. *Information Sciences*, 109(1-4), 1998.

[93] Y. Y. Yao. Constructive and algebraic methods of the theory of rough sets. *Journal of Information Sciences*, 109:21–47, 1998.

[94] Y. Y. Yao. Generalized rough set models. *Rough Sets in Knowledge Discovery*, pages 286–318, 1998.

[95] Y. Y. Yao. Relational interpretations of neighborhood operators and rough set approximation operators. *Information Sciences*, 111(1-4):239–259, 1998.

[96] Y. Y. Yao. On generalizing rough set theory. *LNCS Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing*, 2639:44–51, 2003.

[97] Y. Y Yao. Probabilistic approaches to rough sets. *Expert Systems 20*, pages 287–297, 2003.

[98] Y. Y. Yao. A note on definability and approximations. *LNCS Transactions on 1310 Rough Sets*, 7(LNCS 4400), 2007.

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

[100] Y. Y. Yao and T. Y. Lin. Generalization of rough sets using modal logics. *Intelligent Automation & Soft Computing*, 2(2):103–119, 1996.

[101] 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.

[102] N. Zhong, J.Z Dong, and S. Ohsuga. Data mining: a probabilistic rough set approach. *Rough Sets in Knowledge Discovery 2*, pages 127–146, 1998.

[103] 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.

[104] 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.

[105] W. Zhu. Generalized rough sets based on relations. *Information Sciences*, 177(22):4997–5011, 2007.

[106] W. Zhu. Topological approaches to covering rough sets. *Information Sciences*, 177(6):1499–1508, 2007.

[107] W. Zhu. Relationship between generalized rough sets based on binary relation and covering. *Information Sciences*, 179(3):210–225, 2009.

[108] W. Zhu and F. Wang. Relationships among three types of covering rough sets. *Granular Computing, 2006 IEEE International Conference*, pages 43–48, 2006.

[109] W. Ziarko. Probabilistic rough sets. *Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing, Proceedings of RSFDGrC'05, LNAI 3641*, pages 283–293., 2005.