Brock University

Department of Computer Science

A relational logic for spatial contact based on rough set approximation

Ivo Düntsch, Ewa Orloska, Hui Wang

Brock University
Department of Computer Science
St. Catharines
Ontario Canada L2S 3A1

Institute of Telecommunications
Szachowa 1, 04-894
Warszawa, Poland

School of Computing and Mathematics
University of Ulster at Jordanstown
Newtownabbey, BT 37 0QB, N.Ireland

Technical Report # CS-15-03
November 2015

A relational logic for spatial contact based on rough set approximation

Ivo Düntsch[*]
Brock University
St. Catharines, Ontario, L2S 3A1, Canada
duentsch@brocku.ca

Ewa Orłowska[†]
Institute of Telecommunications
Szachowa 1, 04–894, Warszawa, Poland
orlowska@itl.waw.pl

Hui Wang
School of Computing and Mathematics
University of Ulster at Jordanstown
Newtownabbey, BT 37 0QB, N.Ireland
H.Wang@ulster.ac.uk

November 14, 2015

Abstract

In previous work we have presented a class of algebras enhanced with two contact relation used in spatial reasoning on the basis of rough sets. In this paper we present a relational logic for such structures in the spirit of Rasiowa – Sikorki proof systems.

[*] Ivo Düntsch gratefully acknowledges support by the Natural Sciences and Engineering Research Council of Canada Discovery Grant 250153.

[†] Ewa Orlowska gratefully acknowledges partial support from the National Science Centre project DEC-2011/02/A/HS1/00395.

1 Introduction

In the context of spatial contact, we have proposed a class of algebras which resemble algebras obtained from the approximation operators of rough sets [7]. It was shown that the resulting class was definitionally equivalent to the class of regular double Stone algebras. We used this setup to define approximations for contact relations and part of relations as well, along with their appropriate algebraic and mereological structures. In the present paper we fulfill the promise made in [7] and present relational logics for these systems.

Contact structures may be viewed as an extension of Leśniewski's mereological structures [9] – extensions primarily based on Whitehead's notion of "connection" [20]. A *contact structure* is a triple $\langle U, P, C \rangle$ where P is a partial order on U and C a *contact relation* satisfying 1 – 3 below:

$$C \text{ is reflexive.} \tag{1}$$

$$C \text{ is symmetric.} \tag{2}$$

$$\text{If } C(x) = C(y), \text{ then } x = y. \tag{3}$$

Here, for $z \in W$, we let $C(z) = \{t \in W : zRt\}$. By the *extensionality axiom* (3), a region is determined by all regions to which it is in contact. The elements of U are considered as regions in some geometrical or topological structure. We say that regions x and y *overlap* if there is some z such that zPx and zPy. Usually, the set of regions carries, in addition, an algebraic structure which is compatible with P or C in various degrees, such as Boolean contact algebras, contact lattices etc, see [2] for an overview.

In many cases, regions cannot be exactly determined, but only described by approximations; a case in point are the regions of a computer screen determined by the chosen resolution, membership in which can only be described by a lower and an upper approximation. To model our intuition of approximate region we follow the paradigm of rough sets [15] and make the assumption that there is a collection B of *crisp* or *definable* regions, which forms a Boolean algebra with natural order $\leq$, such that P restricted to B coincides with $\leq$. The crisp regions delineate the bounds up to the granularity of which other regions can be observed. The power of observation is expressed by pairs of the form $\langle a, b \rangle$, $a \leq b$, where a, b are definable regions. In other words, for each (unknown) region x there is a lower bound $i(x) = a$ and an upper bound $h(x) = b$, both of which are crisp, up to which x is discernible. If $i(x) = h(x)$, then x itself is definable. The pair $\langle i(x), h(x) \rangle$ is called an *approximating region*. We also assume that the bounds $\langle i(x), h(x) \rangle$ are best possible; in other words

$$\text{No definable region } c \text{ with } i(x) \lneq c \text{ is a part of } x, \tag{4}$$

$$\text{If } c \text{ is definable and } c \lneq h(x), \text{ then } x \text{ overlaps with } -c. \tag{5}$$

Using some additional assumptions which generalize the widely studied Boolean contact algebras, we define an *approximating algebra* (AA) $\langle L, +, \cdot, 0, 1, i, h, - \rangle$ as a structure of type $\langle 2, 2, 0, 0, 1, 1, 1 \rangle$ such that for all $x, y \in L$,

AA1. $\langle L, +, \cdot, 0, 1 \rangle$ is a bounded distributive lattice.

AA2. i is a dually normal multiplicative interior operator on L.

AA3. h is a normal additive closure operator on L.

AA4. $i(h(x)) = h(x),\ h(i(x)) = i(x)$.

AA5. $i(x) = i(y)$ and $h(x) = h(y)$ imply $x = y$.

AA6. $-$ is a partial unary operator defined on the set $B(L) = \{x \in L | x = h(x)\}$ of closed elements such that

$$x \cdot -x = 0,\ x + -x = 1.$$

Observe that AA1 – AA4 axiomatise the lattice based version of a modal S5 algebra.

The class AA is definitionally equivalent to the class of regular double Stone algebras (see Section 2.1. of [7]), and thus, to the class of subalgebras of full algebras of rough sets [3, 13, 14].

To approximate a contact relation we need two extensions of structures with a crisp contact: An *approximate contact algebra* (ACA) is a structure $\langle L, +, \cdot, 0, 1, i, h, C^i, C^h \rangle$ such that $\langle L, +, \cdot, 0, 1, i, h \rangle$ is an AA and C^i and C^h are binary relations on L such that

$$C^i = C^{i\smile},\ C^h = C^{h\smile} \tag{6}$$

$$1' \subseteq C^i \cap C^h, \tag{7}$$

$$xC^iy \Longleftrightarrow i(x)C^ii(y), \tag{8}$$

$$xC^hy \Longleftrightarrow h(x)C^hh(y), \tag{9}$$

$$h(x)C^ih(y) \Longleftrightarrow h(x)C^hh(y), \tag{10}$$

$$C^h(h(x)) \subseteq C^h(h(y)) \Longleftrightarrow h(x) \leq h(y). \tag{11}$$

For further details we refer the reader to [7].

2 Relational proof systems

In many theories that are being developed in computer science one can observe that there is an increasing tendency to use models based on algebras of relations or modal logics which, as we know today, are closely related to the algebras of relations. Since the invention of Kanger–Kripke semantics, modal logics are viewed as structures expressing an interplay between relations and sets; moreover, modal logics can also be given a purely relational semantics [10, 11]. In this way, any modal logic can be equivalently represented as a class of algebras of relations. However, one can also observe that quite often for the relational or modal models of the respective theories there is a need to introduce algebraic structures which are not exactly standard algebras of relations, but algebras obtained from them by adding or deleting some relational constants and/or operations, or by allowing only some fixed relations as generators and/or by postulating various constraints on these generators.

A general framework for designing proof systems for theories whose models involve relational structures is based on the methodology of relational proof systems in the style of [17]. The systems consist of decomposition rules, specific rules and (sequences of) axiomatic expressions. A decomposition rule when applied to an expression of the theory returns a set of expressions which are syntactically simpler than the original one. These rules provide definitions of relational operators. The specific rules are the counterparts of relational constraints. It is worth mentioning that in the Hilbert-style proof systems for the applied modal logics it is often the case that not all the relational constraints can be explicitly expressed and axiomatised. In an RS system, proving a theorem amounts to expanding a (proof) tree with the formula to be proved as the root, with the aim of closing all branches with axiomatic sequences. Every step of the expansion is based on either a decomposition rule or a specific rule. It should be mentioned that rules in RS systems go in both directions, that is, they preserve and reflect validity of expressions. A transfer of validity from the conclusion of a rule to the premise is needed for the soundness of the system, whereas the other direction is required for completeness.

An important advantage of the relational proof systems is their modularity. A (standard) algebra of relations is a combination of a Boolean algebra and an involuted monoid [8]. Thus, the rules for the Boolean operators and the monoid operators are common to all the relational proof systems. Any particular algebra of relations usually requires some additional rules reflecting its specificity, but the standard part remains unchanged. It can possibly be reduced if, say, some standard operators are not needed, but once it is decided how much of the standard relational resources will be used in a model of the theory under consideration, we can use previously implemented rules.

The experience with relational proof systems designed until now shows that many constraints which are not modally expressible receive an explicit representation in the form of a relational rule or a relational axiomatic sequence. As a case in point, it was shown in [6] that the extensionality axiom (2) of contact relations is not expressible in a classical modal logic, nor, as shown in [5], in its sufficiency counterpart. As another example, the fact that a relation is an intersection of other relations is not expressible in the standard modal language but it is expressible in the form of a relational rule.

It is known that not only modal logics but many other non-classical logics have a purely relational semantics where the formulas are interpreted as relations (as opposed to Kripke semantics which interprets formulas as sets), and hence the relational proof systems are an alternative way of developing deduction system for these logics. For a comprehensive treatment of RS systems we invite the reader to consult [12], where, in particular, Chapter 25 describes the methodology.

For our approximating contact algebras we construct sound and complete relational logics in two steps. First, relational versions of AA and ACA are obtained by expressing operators as relations with suitable arity and properties. In a second step, a sound and complete proof system with these structures as intended models is exhibited.

3 Translating ACAs to relational structures

Following the remarks in the introduction we will first develop a relational version of approximation algebras, where all operations are replaced by appropriate relations.

Recall that there are two equivalent ways of describing a lattice: First, as an algebra satisfying various axioms and the lattice ordering being defined by the operators. Second, one can regard a lattice as a set X endowed with a partial order $\leq$ such that supremum and infimum exist for any two elements of X, and this is the route we shall pursue. Suppose that $\leq$ is a partial order on a set X. We denote the converse of $\leq$ by $\geq$, and set $\lneq \overset{\text{def}}{=} \leq \cap -1'$.

First, we want to express by relations the existence of a supremum and infimum for two elements of X. Consider the following ternary relations on X:

$$S(x,y,z) \iff x \leq z \text{ and } y \leq z \text{ and } (\forall t)[x \leq t \text{ and } y \leq t \text{ implies } z \leq t], \tag{12}$$

$$M(x,y,z) \iff z \leq x \text{ and } z \leq y \text{ and } (\forall t)[t \leq x \text{ and } t \leq y \text{ implies } t \leq z]. \tag{13}$$

Clearly, $S(x,y,z)$ if and only if z is the supremum of x and y, and analogously for $M(x,y,z)$. We need two axioms to ensure existence of supremum and infimum:

$$(\forall x)(\forall y)(\exists z)S(x,y,z), \tag{14}$$

$$(\forall x)(\forall y)(\exists z)M(x,y,z). \tag{15}$$

We also want $\leq$ to be bounded. Thus, we require constants $0_X, 1_X$ in X such that

$$(\forall x) 0_X \leq x, \tag{16}$$

$$(\forall x) x \leq 1_X. \tag{17}$$

Antisymmetry of $\leq$ ensures that the elements 0_X and 1_X are unique. Thus far, $\langle X, \leq, S, M, 0_X, 1_X \rangle$ relationally defines a bounded lattice, which we shall just denote by X if no confusion can arise. As shown in [7], the distributivity laws are expressed as follows:

$$M(x,s,t) \wedge S(y,z,s) \Longleftrightarrow ((M(x,y,u) \wedge M(x,z,v) \Rightarrow S(u,v,t)), \tag{18}$$

$$S(x,s,t) \wedge M(y,z,s) \Longleftrightarrow ((S(x,y,u) \wedge S(x,z,v) \Rightarrow M(u,v,t)). \tag{19}$$

Seeing that, as functions, i and h are binary relations, all that we have to do is to translate the system (AA2) – (AA5) into a relational form; we shall use H and I as relational counterparts for h and i, respectively.

$$I^{\smile} \,;\, I \subseteq 1' \qquad \text{I is functional.} \tag{20}$$

$$I \,;\, V = V \qquad \text{I is total.} \tag{21}$$

$$I \subseteq \geq \qquad \text{I is antitone.} \tag{22}$$

$$I^{\smile} \,;\, \leq \,;\, I \subseteq \leq \qquad \text{I preserves order.} \tag{23}$$

$$I \,;\, I \subseteq 1' \qquad \text{I is idempotent.} \tag{24}$$

$$1I1 \qquad \text{I is dually normal.} \tag{25}$$

$$xIy \wedge sIt \wedge M(x,s,p) \wedge M(y,t,q) \Rightarrow pIq \qquad \text{I is multiplicative.} \tag{26}$$

Thus far, we have expressed (AA2), i.e. that I is a multiplicative dually normal interior operator. Translating (AA3) we obtain

$$H^{\smile} \,;\, H \subseteq 1' \qquad \text{H is functional.} \tag{27}$$

$$H \,;\, V = V \qquad \text{H is total.} \tag{28}$$

$$H \subseteq \leq \qquad \text{H is monotone.} \tag{29}$$

$$H^{\smile} \,;\, \leq \,;\, H \subseteq \leq \qquad \text{H preserves order.} \tag{30}$$

$$H \,;\, H \subseteq 1' \qquad \text{H is idempotent.} \tag{31}$$

$$0H0 \qquad \text{H is normal.} \tag{32}$$

$$xHy \wedge sHt \wedge S(x,s,p) \wedge S(y,t,q) \Rightarrow pHq \qquad \text{H is additive.} \tag{33}$$

The equations (AA4) and (AA5) become, respectively,

$$(H\ ;\ I)\cup(I\ ;\ H)\subseteq 1', \tag{34}$$

$$(H\ ;\ H^{\smile})\cap(I\ ;\ I^{\smile})\subseteq 1'. \tag{35}$$

For (AA6) we need to translate the conditions of the complementation $-$ on $B(L)$ which we will denote by D. Thus, let

$$xDy \iff S(x,y,1) \text{ and } M(x,y,0). \tag{36}$$

The axiom which assures that closed elements (and only those) have a complement is

$$(D\ ;\ V)\cap 1' = (H^{\smile}\ ;\ V)\cap 1'. \tag{37}$$

This says that the domain of D is equal to the range of H. Note that D is functional, since L is distributive, and thus, each $x\in L$ has at most one complement.

After this preparation we define an *approximating frame* (AF) as a structure

$$\langle X,\leq,S,M,I,H,1',D,0_X,1_X\rangle$$

such that

1. $\langle X,\leq\rangle$ is a partially ordered set.
2. S and M are ternary relations of X defined by (12) and (13) which satisfy (14) and (15).
3. 0_X and 1_X are elements of X which satisfy (16) and (17).
4. $1'$ is the identity relation on X.
5. I and H are binary relations on X which satisfy (20) – (35).
6. D is a binary relation on X which satisfies (36) and (37).

From the definitions, the following result is now immediate:

Proposition 3.1. *1. Let $\langle L,\vee,\wedge,i,h,-,0,1\rangle$ be an AA, and let*

$$X = L,\ 0_X = 0,\ 1_X = 1,$$
$$S(x,y,z) \iff z = x\vee y,\ M(x,y,z) \iff z = x\wedge y,$$
$$xIy \iff y = i(x),\ xHy \iff y = h(x),$$
$$xDy \iff x\in B(L) \textit{ and } y = -x.$$

Then, $\langle X,\leq,S,M,I,H,1',D,0_X,1_X\rangle$ is an AF.

2. *Let $\langle X,\leq,S,M,I,H,1',D,0_X,1_X\rangle$ be an AF, and set $L = X$, $0 = 0_X, 1 = 1_X$. For $x,y \in L$ let $x \vee y$ be the unique z with $S(x,y,z)$, and $x \wedge y$ be the unique z with $I(x,y,z)$. Define $h : X \to L$ by letting $h(x)$ be the unique $y \in X$ with xHy, and $i : X \to L$ by letting $i(x)$ be the unique $y \in X$ with xIy. Furthermore, for $x \in \mathrm{ran}(H)$, let $-x$ be the unique $y \in \mathrm{ran}(H)$ with xDy. Then, $\langle L,\vee,\wedge,i,h,-,0,1\rangle$ is an AA.*

3. *With these translations, the classes of AAs and AFs are mutually interpretable in each other.*

Next, we add the translations for the contact relations: An *approximating contact frame* (ACF) is a structure $\langle X,\leq,S,M,I,H,1',D,0_X,1_X,C^I,C^H\rangle$ such that $\langle X,\leq,S,M,I,H,1',D,0_X,1_X\rangle$ is an AF, C^I and C^H are binary relations on L for which the following hold:

$$C^I = C^{I\smile},\ C^H = C^{H\smile}, \tag{38}$$

$$1' \subseteq C^I \cap C^H, \tag{39}$$

$$C^I = I \ ; \ C^I \ ; \ I^{\smile}, \tag{40}$$

$$C^H = H \ ; \ C^H \ ; \ H^{\smile}, \tag{41}$$

$$H \ ; \ C^I \ ; \ H^{\smile} = C^H, \tag{42}$$

$$-(H \ ; \ C^H) \setminus_{\mathrm{res}} -(H \ ; \ C^H) = H \ ; \ \leq \ ; \ H^{\smile}. \tag{43}$$

Here, $Q \setminus_{\mathrm{res}} R$ is the *residual of R by Q*, defined as

$$x(Q \setminus_{\mathrm{res}} R)y \text{ if and only if } (\forall z)(yQz \text{ implies } xRz). \tag{44}$$

The conditions (38) – (43) are straightforward adaptations of (6) – (11) which is easily seen.

4 The rules

The rules are formulated in the language of algebras of binary relations generated from the relations $\leq, I, H, 1', D, O_X, 1_X$ with the operations of union, intersection, composition, complement, and the language of ternary relations generated from S,M with the operations of union, intersection, and complement. We will define a proof system for the model class ACF in the style of [17] (RS system). As the underlying language is clear from the rules, we will not expand on these here. The general structure of such languages and many examples can be found in [12].

There are two types of rules:

1. *Decomposition rules* break up formulas into an equivalent sequence of formulas with a simpler structure.
2. *Specific rules* modify a sequence of formulas, and have the status of structural rules.

In most instances, the rules are actually rule schemas. The role of axioms is played by *axiomatic sequences*. In an RS system one attempts to verify a formula by closing the branches of its decomposition tree with axiomatic sequences. Rules in RS systems go in both directions: we call a rule *correct*, if

> *The upper sequence is valid if and only if the lower sequence(s) is (are) valid.*

Here, a sequence of formulas is valid if its meta-level disjunction is valid.

The decomposition rules for the binary relational operations are given in Table 1, the specific rules for lattice properties are given in Table 2, the AF specific rules for I are given in Table 3, the AF specific rules for H are given in Table 4, the AF specific rules for D are given in Table 5, and the specific rules for the contact relations are given in Table 6. The decomposition rules for ternary relations with respect to $\cup, \cap, -$ and $-\cup, -\cap, --$ are analogous to those for binary relations.

$(\cup)$	$\dfrac{\Gamma, x(R\cup Q)y, \Delta}{\Gamma, xRy, xQy, \Delta}$	$(-\cup)$	$\dfrac{\Gamma, x-(R\cup Q)y, \Delta}{\Gamma, x(-R)y, \Delta \mid \Gamma, x(-Q)y, \Delta}$
$(\cap)$	$\dfrac{\Gamma, x(R\cap Q)y, \Delta}{\Gamma, xRy, \Delta \mid \Gamma, xQy, \Delta}$	$(-\cap)$	$\dfrac{\Gamma, x-(R\cap Q)y, \Delta}{\Gamma, x(-R)y, x(-Q)y, \Delta}$
$(^{\smile})$	$\dfrac{\Gamma, xR^{\smile}y, \Delta}{\Gamma, yRx, \Delta}$	$(-^{\smile})$	$\dfrac{\Gamma, x(-R^{\smile})y, \Delta}{\Gamma, y(-R)x, \Delta}$
$(--)$	$\dfrac{\Gamma, x(--R)y, \Delta}{\Gamma, xRy, \Delta}$		
$(;)$	$\dfrac{\Gamma, x(R;Q)y, \Delta}{\Gamma, xRz, \Delta, x(R;Q)y \mid \Gamma, zQy, \Delta, x(R;Q)y}$		
$(-;)$	$\dfrac{\Gamma, x-(R;Q)y, \Delta}{\Gamma, x(-R)z, z(-Q)y, \Delta}$	where z is a new variable	

Table 1: Decomposition rules for binary relations

(refl $\leq$)	$\dfrac{\Gamma, x \leq y, \Delta}{\Gamma, x1'y, \Delta, x \leq y}$	
(trans $\leq$)	$\dfrac{\Gamma, x \leq y, \Delta}{\Gamma, x \leq z, \Delta, x \leq y \mid \Gamma, z \leq y, \Delta, x \leq y}$	
(antisym $\leq$)	$\dfrac{\Gamma, x1'y, \Delta}{\Gamma, x \leq y, \Delta \mid \Gamma, y \leq x, \Delta}$	
(S)	$\dfrac{\Gamma, S(x,y,z), \Delta}{\Gamma, x \leq z, \Delta \mid \Gamma, y \leq z, \Delta \mid \Gamma, x(- \leq)t, y(- \leq)t, z \leq t, \Delta}$	z, t new
(-S)	$\dfrac{\Gamma, -S(x,y,z), \Delta}{\begin{array}{c} x- \leq z, y- \leq z, x \leq t, \Delta, -S(x,y,z) \mid \\ x- \leq z, y- \leq z, y \leq t, \Delta, -S(x,y,z) \mid x- \leq z, y- \leq z, z \leq t, \Delta, -S(x,y,z) \end{array}}$	
(M)	$\dfrac{\Gamma, M(x,y,z), \Delta}{\Gamma, z \leq x, \Delta \mid \Gamma, z \leq y, \Delta \mid \Gamma, t(- \leq)x, t(- \leq)y, t \leq z, \Delta}$	z, t new
(-M)	$\dfrac{\Gamma, -M(x,y,z), \Delta}{\begin{array}{c} x- \geq z, y- \geq z, x \geq t, \Delta, -M(x,y,z) \mid \\ x- \geq z, y- \geq z, y \geq t, \Delta, -M(x,y,z) \mid x- \geq z, y- \geq z, z \geq t, \Delta, -M(x,y,z) \end{array}}$	
(18) ⇒	$\dfrac{\Gamma}{\Gamma, M(x,s,t) \mid \Gamma, M(x,y,u) \mid \Gamma, M(x,z,v) \mid \Gamma, S(y,z,s) \mid \Gamma, -S(u,v,t)}$	
(18) ⇐	$\dfrac{\Gamma}{\Gamma, -M(x,y,u), -M(x,z,v), S(u,v,t) \mid \Gamma, -M(x,s,t), -S(y,z,s)}$	
(19) ⇒	$\dfrac{\Gamma}{\Gamma, S(x,s,t) \mid \Gamma, S(x,y,u) \mid \Gamma, S(x,z,v) \mid \Gamma, M(y,z,s) \mid \Gamma, -M(u,v,t)}$	
(19) ⇐	$\dfrac{\Gamma}{\Gamma, -S(x,y,u), -S(x,z,v), M(u,v,t) \mid \Gamma, -S(x,s,t), -M(y,z,s)}$	
$Q \subseteq R$	$\dfrac{xRy}{xQy, xRy}$	
$Q \backslash_{\mathrm{res}} R$	$\dfrac{Q \backslash_{\mathrm{res}} R}{y(-Q)z, xRz}$	

Table 2: Specific rules for lattice properties

$$(20) : \frac{\Gamma, x1'y, \Delta}{\Gamma, zIx, \Delta, x1'y \mid \Gamma, zIy, x1'y} \tag{45}$$

$$(21) : \frac{\Gamma}{\Gamma, x - Iz} \qquad z \text{ new}, \tag{46}$$

$$(22) : \frac{\Gamma, x \leq y, \Delta}{\Gamma, yIx, \Delta, x \leq y} \tag{47}$$

$$(23) : \frac{\Gamma, x \leq y, \Delta}{\Gamma, zIx, \Delta, x \leq y \mid \Gamma, x \leq t, \Delta, x \leq y \mid \Gamma, tIy, \Delta, x \leq y} \tag{48}$$

$$(24) : \frac{\Gamma, x1'y, \Delta}{\Gamma, xR_I z, \Delta, x1'y \mid \Gamma, zIy, \Delta, x1'y} \tag{49}$$

$$(26) : \frac{pIq}{xIy, pIq \mid sIt, pIq \mid M(x,s,p), pIq \mid M(y,t,q), pIq} \tag{50}$$

Table 3: Specific rules for *I*

$$(27) : \frac{\Gamma, x1'y, \Delta}{\Gamma, zHx, \Delta, x1'y \mid \Gamma, zHy, x1'y} \tag{51}$$

$$(28) : \frac{\Gamma}{\Gamma, x - Hz} \qquad z \text{ new}, \tag{52}$$

$$(29) : \frac{\Gamma, x \leq y, \Delta}{\Gamma, xHy, \Delta, x \leq y} \tag{53}$$

$$(30) : \frac{\Gamma, x \leq y, \Delta}{\Gamma, zHx, \Delta, x \leq y \mid \Gamma, x \leq t, \Delta, x \leq y \mid \Gamma, tHy, \Delta, x \leq y} \tag{54}$$

$$(31) : \frac{\Gamma, x1'y, \Delta}{\Gamma, xR_H z, \Delta, x1'y \mid \Gamma, zHy, \Delta, x1'y} \tag{55}$$

$$(33) : \frac{pHq}{xHy, pHq \mid sHt, pHq \mid S(x,s,p), pHq \mid S(y,t,q), pHq} \tag{56}$$

Table 4: Specific rules for *H*

$$(36) \Rightarrow: \quad \frac{\Gamma}{x(-D)y, \Gamma \mid S(x,y,1), \Gamma \mid M(x,y,0), \Gamma} \tag{57}$$

$$(36) \Leftarrow: \quad \frac{xDy}{S(x,y,1), xDy | M(x,y,0), xDy} \tag{58}$$

$$(37) \subseteq: \quad \frac{\Gamma}{zHx, \Gamma \, | x1'y, \Gamma \mid x(-D)t, x(-1')y\Gamma} \qquad t \text{ new} \tag{59}$$

$$(37) \supseteq: \quad \frac{\Gamma}{zDx, \Gamma \, | x1'y, \Gamma \mid t(-H)x, x(-1')y\Gamma} \qquad t \text{ new} \tag{60}$$

Table 5: Specific rules for D

$$(38): \quad \frac{\Gamma,\ xC^I y,\ \Delta}{\Gamma,\ yC^I x,\ \Delta}, \frac{\Gamma,\ xC^H y,\ \Delta}{\Gamma,\ yC^H x,\ \Delta} \tag{61}$$

$$(39): \quad \frac{\Gamma,\ x(-1')y,\ \Delta}{\Gamma,\ x(-C^I)y,\ x(-C^H)y,\ \Delta,\ x(-1')y} \tag{62}$$

$$(40): \quad \frac{\Gamma,\ x(-C^I)y,\ \Delta}{\Gamma,\ x(-I)u,\ y(-I)w,\ u(-C^I)w,\ \Delta,\ x(-C^I)y}, \qquad u,w \text{ new, } u \neq w, \tag{63}$$

$$\frac{\Gamma,\ xC^I y,\ \Delta}{\Gamma,\ xIu,\ \Delta,\ xC^I y \mid \Gamma,\ uC^I w,\ \Delta,\ xC^I y \mid \Gamma,\ yIw,\ \Delta,\ xC^I y} \tag{64}$$

$$(41): \quad \frac{\Gamma,\ x(-C^H)y,\ \Delta}{\Gamma,\ x(-H)u,\ y(-H)w,\ u(-C^H)w,\ \Delta,\ x(-C^H)y}, \qquad u,w \text{ new, } u \neq w, \tag{65}$$

$$\frac{\Gamma,\ xC^H y,\ \Delta}{\Gamma,\ xHu,\ \Delta,\ xC^H y \mid \Gamma,\ uC^H w,\ \Delta,\ xC^H y \mid \Gamma,\ yHw,\ \Delta,\ xC^H y} \tag{66}$$

$$(42): \quad \frac{\Gamma,\ xC^H y,\ \Delta}{\Gamma,\ xHu,\ \Delta,\ xC^H y \mid \Gamma,\ uC^I w,\ \Delta,\ xC^H y \mid \Gamma,\ yHw,\ \Delta,\ xC^H y} \tag{67}$$

$$\frac{\Gamma,\ x(-C^H)y,\ \Delta}{\Gamma,\ x(-H)u,\ y(-H)w,\ u(-C^I)w, \Delta, x(-C^H)y} \qquad u,w \text{ new, } u \neq w \tag{68}$$

The next two rules define the auxiliary relation $R = H \ ; \ C^H$ used for (43)

$$\frac{\Gamma,\ x(-R)y,\ \Delta}{\Gamma,\ x(-H)u,\ u(-C^H)y,\ \Delta,\ x(-R)y} \qquad u \text{ new} \tag{69}$$

$$\frac{\Gamma,\ xRy,\ \Delta}{\Gamma,\ xHu,\ \Delta,\ xRy \mid \Gamma,\ uC^H y,\ \Delta, xRy} \tag{70}$$

$$(43): \quad \frac{\Gamma}{\Gamma,\ yRz,\ x(-R)z \mid \Gamma,\ x(-H)u,\ u(-\leq)w,\ y(-H)w} \qquad u,w,z \text{ new, distinct} \tag{71}$$

(72)

Table 6: Specific rules for C^I and C^H

5 Soundness

The rules of our system are defined according to some general principles which tell us how to assign a rule to a scheme of the condition that the underlying relations are assumed to satisfy. The properties of the lattice order $\leq$ as well as the conditions (20), …, (24), (27), …, (31), (34), (35), (37), and (38) – (42) are instances of such schemes, which we will now describe. By a *relational literal* we mean an expression of one of the forms $R, -R, R^{\smile}$, where R is a relational variable or constant.

$$\text{(s1).}\quad R_1;\ldots;R_n \subseteq Q_1;\ldots;Q_m.$$

$$\text{(s2}\subseteq\text{).}\quad -(R_1;\ldots;R_n) \subseteq Q_1;\ldots;Q_m,\ \text{ i.e. } (R_1;\ldots;R_n) \cup (Q_1;\ldots;Q_m) = V.$$

$$\text{(s2}\supseteq\text{).}\quad -(R_1;\ldots;R_n) \supseteq Q_1;\ldots;Q_m,\ \text{ i.e. } (R_1;\ldots;R_n) \cap (Q_1;\ldots;Q_m) = -V.$$

$$\text{(s3).}\quad (R^1_1;\ldots;R^1_{f(1)}) \cup \cdots \cup (R^n_1;\ldots;R^n_{f(n)}) \subseteq P.$$

$$\text{(s4).}\quad (R^1_1;\ldots;R^1_{f(1)}) \cap \cdots \cap (R^n_1;\ldots;R^n_{f(n)}) \subseteq P.$$

Here, $n,m \geq 1$, and P, R_i, Q_j etc are relational literals.

The conditions (26) and (33) which involve both binary and ternary relations are formulated in a first order language and have the following form:

$$\text{(s5).}\quad (\forall x_1,\ldots,x_n)[F_1 \wedge \cdots \wedge F_m \implies F].$$

Here, $n,m \geq 1$, and $F_1,\ldots F_m, F$ are first order literals (i.e. atomic formulas or negations of atomic formulas), whose variables are among $x_1,\ldots,x_n$. For the sake of simplicity we present the schemes for the rules corresponding to the conditions above only for the case $n = m = 2$.

$$\text{(rs1).}\quad \frac{K, x(-R_1)z, z(-R_2)y, H}{K, x(-Q_1)t, t(-Q_2)y, H, x(-R_1)z, z(-R_2)y}$$

$$\text{(rs2}\subseteq\text{).}\quad \frac{K}{K, x(-R_1)z, z(-R_2)y,\ \mid K, x(-Q_1)t, t(-Q_2)y},\quad z,t \text{ new}$$

$$\text{(rs2}\supseteq\text{).}\quad \frac{K, x(-R_1)t, t(-R_2)y, H}{K, xQ_1z, H, x(-R_1)t, t(-R_2)y \mid K, zQ_2y, H, x(-R_1)t, t(-R_2)y}$$

$$\text{(rs3).}\quad \frac{K, xPy, H}{K, xR_1z, xQ_1t, H, xPy \mid K, zR_2y, xQ_1t, H, xPy \mid K, xR_1z, xQ_2t, H, xPy \mid K, zR_2y, xQ_2t, H, xPy}$$

$$\text{(rs4).}\quad \frac{K,xPy,H}{K,xR_1z,H,xPy \mid K,zR_2y,H,xPy \mid K,xQ_1t,H,xPy\mid K,tQ_2y,H,xPy}$$

$$\text{(rs5).}\quad \frac{K,F,H}{K,F_1,H,F \mid \ldots \mid K,F_n,H,F}$$

A particular case of ((s1)) is

(s1'). $R;1 = 1$.

We will treat this case separately, since a constraint of this form is often present in relational theories. The corresponding rule is

$$\text{(rs1').}\quad \frac{K}{K,x(-R)y}, \quad y \text{ new}$$

We now have the following result:

Proposition 5.1. *1. For $i = 1,\ldots 5$, a rule (rsi) is correct if and only if (si) is satisfied in every ACF.*

2. The axiomatic sequences are valid.

For lack of space we omit the proofs and refer the reader to [12] for application of the general cases.

6 Completeness

Owing to the large number of rules and the limitations on space we shall just outline the general procedure to show completeness. For a logic $\mathscr{L}$, an *$\mathscr{L}$– proof tree* for a formula φ is a tree such that each of its nodes is a set of formulas semantically interpreted as a disjunction of its elements and constructed as

1. φ is at the root of the tree.
2. Each node except the root is obtained by an application of a rule to its predecessor node.

3. A node does not have successors whenever its set of formulas is axiomatic or none of the rules is applicable to its set of formulas.

A branch of an $\mathscr{L}$ – proof tree is *complete* whenever all the rules applicable to its nodes have been applied. An $\mathscr{L}$ – proof tree is called complete whenever all of its branches are complete. A branch is called closed whenever it contains a node with an axiomatic set of formulas. A branch is said to be open whenever it is complete and non-closed.

For an open branch b of a complete proof tree, a *branch structure* is a structure $\langle X^b, m^b \rangle$ such that its universe is the set of object symbols (individual variables + constants) of $\mathscr{L}$ and for every binary relational symbol R, $m^b(R) = \{\langle x, y \rangle : xRy \notin b\}$, similarly for other arities.

To prove completeness of a proof system we need to show the following three facts:

Closed branch property For every branch of an $\mathscr{L}$ – proof tree, if φ and $\neg\varphi$ for an atomic formula φ belong to a branch, then the branch can be closed, that is both of these formulas appear in the same node of the branch.

Branch model property Let b be an open branch of a complete proof tree. Then the branch structure M^b is an $\mathscr{L}$– model, referred to as a branch model, that is, it satisfies all the constraints assumed in the models of $\mathscr{L}$.

Satisfaction in branch model property Let b be an open branch of a complete $\mathscr{L}$– proof tree and let v^b be the identity valuation in M^b. Then, for every formula φ of $\mathscr{L}$, if v^b satisfies φ in M^b, then $\varphi \notin b$.

7 Implementation

We have implemented the AA logic in Prolog[1]. In the RS proof system, proving a theorem amounts to expanding a tree with the formula to be proved as the root, with the aim of closing all branches with axiomatic sequences. Every step of the expansion is based on either the decomposition rules or the specific rules. The number of rules is fairly large (57 altogether) and hence the search space is huge. Therefore we need some heuristics to reduce the search space to a manageable level. Since our implementation is intended as a general theorem proving program, the heuristics employed should therefore be general enough in the sense that they only prune unnecessary branches.

[1] The source code is available from the third author.

1. Decomposition/specific heuristic: apply decomposition rules before specific rules for any node (sequence) in the proof tree.
2. Whole/part heuristic: for any decomposed sequence (node), we first of all take the sequence as a single entity and apply the distributive specific rules; then we look at the individual formula in the sequence by applying other rules.
3. Specific/general heuristic: Some rules are specific to a particular algebra (e.g., contact algebra) while some rules are general in that they are applicable to all algebras in the class. In this approach, we apply less general rules before more general rules.
4. Non-repetitive heuristic: no rule is applied repetitively in any path in the proof tree.

Having these heuristics in mind we designed the following procedure for our implementation.

- Take the formula to be proved as the single root node
- For any node in the tree
  1. Repetitively apply decomposition rules to the node until it can not be decomposed any further. This in effect expands the node into a number of new nodes, thus having branched the proof tree.
  2. For each decomposed node we apply the specific – general heuristic, then the whole – part heuristic. The non-repetitive heuristic is observed throughout the rest of the proof process.

8 Conclusion and outlook

In this paper we have presented a relational logic for approximate contact relations and their algebras. These are a generalisation of Boolean contact algebras [19, 4], and were first studied in [7]; similar work was presented in [18]. Our proof system was given in the style of Rasiowa and Sikorski [17]. For lack of space, we have only indicated how soundness and completeness can be proved; details of the procedures can be found in [12]. In future work we shall investigate approximate proof systems for contact relations which are not necessarily extensional. In this case, the mereological part–of relation is not definable from the contact relation, and additional considerations are required. In this context, we shall also explore the connection to rough mereology [16].

References

[1] Andréka, H., Monk, J. D., Németi, I., Eds.: *Algebraic Logic*, vol. 54 of *Colloquia Mathematica Societatis János Bolyai*, North Holland, Amsterdam, 1991, ISBN 0-444-88543-9.

[2] Cohn, A. G., Hazarika, S. M.: Qualitative spatial representation and reasoning: An overview, *Fundamenta Informaticae*, **46**, 2001, 1–29.

[3] Comer, S.: On connections between information systems, rough sets, and algebraic logic, in: *Algebraic Methods in Logic and Computer Science* (C. Rauszer, Ed.), vol. 28 of *Banach Center Publications*, Polish Academy of Science, Warszawa, 1993, 117–124.

[4] Dimov, G., Vakarelov, D.: Contact algebras and region–based theory of space: A proximity approach – I,II, *Fundamenta Informaticae*, **74**, 2006, 209 – 282.

[5] Düntsch, I., Orłowska, E.: Mixing modal and sufficiency operators, *Bulletin of the Section of Logic, Polish Academy of Sciences*, **28**(2), 1999, 99–106.

[6] Düntsch, I., Orłowska, E.: A proof system for contact relation algebras, *Journal of Philosophical Logic*, **29**, 2000, 241–262.

[7] Düntsch, I., Orłowska, E., Wang, H.: Algebras of approximating regions, *Fundamenta Informaticae*, **46**, 2001, 71–82.

[8] Jónsson, B.: The theory of binary relations, in: Andréka et al. [1], 245–292.

[9] Leśniewski, S.: Grundzüge eines neuen Systems der Grundlagen der Mathematik, *Fundamenta Mathematicae*, **14**, 1929, 1–81.

[10] Orłowska, E.: Relational interpretation of modal logics, in: Andréka et al. [1], 443–471.

[11] Orłowska, E.: Relational Semantics for Non-classical Logics: Formulas are Relations., in: *Philosophical Logic in Poland* (J. Wolenski, Ed.), Kluwer, 1994, 167–186.

[12] Orłowska, E., Golińska-Pilarek, J.: *Dual Tableaux: Foundations, Methodology, Case Studies*, vol. 33 of *Trends in Logic*, Springer Verlag, 2011.

[13] Pagliani, P.: Rough Sets Theory and Logic-Algebraic Structures, in: *Incomplete Information – Rough Set Analysis* (E. Orłowska, Ed.), Physica – Verlag, Heidelberg, 1998, ISBN 3-7908-1049-5, 109–190.

[14] Pagliani, P., Chakraborty, M.: *A Geometry of Approximation*, Springer Verlag, 2008.

[15] Pawlak, Z.: Rough sets, *Internat. J. Comput. Inform. Sci.*, **11**, 1982, 341–356.

[16] Polkowski, L., Skowron, A.: Rough Mereology: A new paradigm for approximate reasoning, *International Journal of Approximate Reasoning*, **15**, 1996, 333–365.

[17] Rasiowa, H., Sikorski, R.: On the Gentzen Theorem, *Fundamenta Mathematicae*, **48**, 1960, 47–69.

[18] Roy, A. J., Stell, J. G.: Spatial relations between indeterminate regions, *Int. J. Approx. Reasoning*, **27**(3), 2001, 205–234.

[19] Stell, J.: Boolean connection algebras: A new approach to the Region Connection Calculus, *Artificial Intelligence*, **122**, 2000, 111–136.

[20] Whitehead, A. N.: *Process and reality*, MacMillan, New York, 1929.