Expressive Power and Complexity in Algebraic Logic

Robin Hirsch*

March 18, 1997

Abstract

Two complexity problems in algebraic logic are surveyed: the satisfaction problem and the network satisfaction problem. Various complexity results are collected here and some new ones are derived. Many examples are given. The network satisfaction problem for most cylindric algebras of dimension four or more is shown to be intractable. Complexity is tied-in with the expressivity of a relation algebra. Expressivity and complexity are analysed in the context of homogeneous representations.

The model-theoretic notion of interpretation is used to generalise known complexity results to a range of other algebraic logics. In particular a number of relation algebras are shown to have intractable network satisfaction problems.

1 Introduction

A basic problem in theoretical computing and applied logic is to select and evaluate the ideal formalism to represent and reason about a given application. Many different formalisms are adopted: classical first-order logic, modal and temporal logics (either propositional or predicate), relational languages, algebraic systems and others. The arguments for and against each of these formalisms has raged for quite some time now, but the criteria for making an objective evaluation have not always been entirely clear.

The issues of complexity (including decidability) and expressive power ought to be fundamental to this debate. If a formalism is undecidable then exact results cannot generally be obtained by a computer, even in principle. Even if it is decidable, if the complexity is high — NP hard say — then in practice the formalism may not be able to give exact solutions unless the problem size is very small. On the other hand, if the expressive power of the language is poor then it may not even be able to represent the problem under consideration. In many cases the trick is to find the optimal balance between expressive power and complexity.

In this paper we will be looking at the expressive power and complexity of certain algebraic logics: cylindric and relation algebras. These algebras were invented primarily to handle algebraically the study of relations of various ranks. Tarski showed [TG87] that relation algebra can act as a vehicle for set theory and hence all of mathematics. Indeed algebraic logic has turned out to have very powerful applications through much of computer science [All81, AK83b, AK83a, All84, AH85a, KV86, AH85b, Pel88, Koo89, VKvB89, KL91, LM93, LR93, Hir94a, Hir96, Hir95] and applies to any system that has to handle relations in a non-trivial way. Thus results about the expressive power and complexity of algebraic logics will have wide repercussions in computer science.

The outline of this article is as follows. First we give the basic definitions of relation algebra, cylindric algebra and the representations of these algebras. We also define a network and consider the question as to whether a network embeds in some representation of an algebra — the so-called network satisfaction problem (NSP). The network satisfaction problem is an example of the

*Supported by SERC grant reference:GR/H46343. Thanks to Ian Hodkinson for correcting important errors in the original draft.

constraint satisfaction problem (CSP) but sufficiently general to be able to deal with an arbitrary CSP. So here again we see that results obtained for the NSP have applications far beyond.

This is followed in section 3 where we give a lower bound to the complexity of the satisfaction problem and give many examples of relation algebras together with some of the known complexity results about them, including some new results obtained here. We also consider the NSP over the minimal four-dimensional cylindric algebra $\mathcal{D}_4$ and show that the complexity of this problem is NP-complete.

Section 4 considers homogeneous representations and gives examples of relation algebras with homogeneous representations and one which has no homogeneous representation. We show that quantifiers can be eliminated in homogenous representations.

In section 5 we expound the notion of an *interpretation* of one algebraic logic in another. This is used to define the expressivity of a relation algebra (or cylindric algebra) and various examples of interpretations are given. There is a basic lemma (lemma 19) which shows that increasing the expressive power of an algebra makes the complexity of the NSP at least as bad.

Interpretations are used in section 6 to show how to reduce a satisfaction problem for one algebra to that of another. For the network satisfaction problem a special interpretation where formulas are defined by networks is used. The intractability of the NSP for $\mathcal{D}_4$ is generalised using interpretations to show that most representable cylindric algebras of dimension four or more have intractable NSPs. Relation algebras can be interpreted in cylindric algebras and this is done to prove the intractability of two relation algebras.

2 Preliminaries

The results about expressive power and complexity later in this paper apply to various algebraic logics. We deal with the two most important kinds: relation algebra and cylindric algebra.

2.1 Relation Algebra

Relation algebras were designed to handle binary relations in an algebraic way. Let us first define a *proper relation algebra* — a concrete structure with binary relations — and then give the definition of an abstract relation algebra.

Definitions

- A *field of sets* F is a set of subsets of some domain X such that F contains the empty set, F contains some biggest set 1_F (not necessarily equal to X) and F is closed under finite unions and taking complements relative to 1_F.
- A *proper relation algebra* (**PRA**)[1] is a domain (D) together with a field of binary relations (B) over D. B must form a field of sets (but note that the top element is not necessarily equal to $D \times D$) including the identity relation ($= \{(d,d) : d \in D\}$) and it must be closed under the operations of taking converse and composition. Just in case these operations are not familiar, the converse of a binary relation r (written $r^{\smile}$) is defined to be $\{(d,e) : (e,d) \in r\}$ and the composition of two binary relations r, s (written $r;s$) is defined to be $\{(d,e) : \exists x \in D, (d,x) \in r \wedge (x,e) \in s\}$.

Now we move on to the algebraic approach. The algebraic counterpart of a field of sets is a boolean algebra and it is a theorem [Sto36] that every boolean algebra is isomorphic to a field of sets. For proper relation algebras, the algebraic counterpart is called a relation algebra, though the correspondence between relation algebras and proper relation algebras is not as close.

[1] We may use **PRA** either as an abbreviation for 'proper relation algebra' or to stand for the class of all proper relation algebras.

Definition An (abstract) relation algebra $\mathcal{A}$ (**RA**) is a tuple $(A, \vee, -, 0, 1, ;, {}^{\smile}, Id)$ which satisfies the following axioms, essentially due to Tarski. For all $a, b, c \in \mathcal{A}$,

1. $(A, \vee, -, 0, 1)$ is a Boolean algebra (1 is the universal element) so we can introduce $\wedge, \leq$ as the usual abbreviations
2. ; is an associative binary operator on $\mathcal{A}$
3. $(a^{\smile})^{\smile} = a$
4. $Id; a = a; Id = a$
5. $a; (b \vee c) = a; b \vee a; c$
6. $(a \vee b)^{\smile} = a^{\smile} \vee b^{\smile}$
7. $(a - b)^{\smile} = a^{\smile} - b^{\smile}$
8. $(a; b)^{\smile} = b^{\smile}; a^{\smile}$
9. $(a; b) \wedge c^{\smile} = 0 \Leftrightarrow (b; c) \wedge a^{\smile} = 0$ [triangle axiom].

An *atom* of $\mathcal{A}$ is a minimal non-zero element under the ordering $\leq$. The set of all atoms of $\mathcal{A}$ is denoted $At(\mathcal{A})$.

Note that $-$ here is a unary operation, but we can define the binary operation $a - b \stackrel{\text{def}}{=} -(b \vee -a)$. In a **PRA** the operation $-$ is interpreted as complement *relative to the top element* 1. These axioms are clearly sound over **PRA**, but they turn out not to be complete [Lyn50] — there are (even finite) relation algebras which are not isomorphic to any proper relation algebra. Let us define a *representation* (X, D) of $\mathcal{A}$ to be an isomorphism X from $\mathcal{A}$ to some proper relation algebra $\mathcal{P}$ with domain D i.e. a bijection from the elements of $\mathcal{A}$ to the binary relations in $\mathcal{P}$ over D and X must respect all the operations. For any representation, it is always the case that $X(1)$ is an equivalence relation over D. This follows from the equations $Id \leq 1$, $1^{\smile} = 1$ and $1; 1 = 1$ which are consequences of axioms 1 to 9. If $X(1) = D \times D$ then call (X, D) a *square* representation. Lyndon's result shows that not every relation algebra is representable. It has been shown [Mon64] that no finite set of axioms can be sound and complete over **PRA**.

Let $(X, D), (Y, E)$ be representations of a relation algebra $\mathcal{A}$. A *base-isomorphism* $h : (X, D) \to (Y, E)$ is a bijection from D to E preserving the relations i.e. for all $d, d' \in D$, for all $a \in \mathcal{A}$, $(d, d') \in X(a)$ if and only if $(h(d), h(d')) \in Y(a)$.

Notation As is standard in model theory (though perhaps not in algebraic logic) we reduce notational clutter by letting the same symbol X stand for the map, the domain of a representation and the name of the representation itself. Thus $x \in X$ means that x is a point in the domain of the representation and for $a \in \mathcal{A}$, $X(a)$ is the binary relation corresponding to a. These different uses are to be interpreted according to their context. If $\bar{x}$ is an n-tuple of elements from the representation X we write $\bar{x} \in X$ instead of $\bar{x} \in X^n$.

Simplicity $\mathcal{A}$ is called *simple* if any homomorphism either is an isomorphism or it maps $\mathcal{A}$ to the trivial relation algebra where $0 = 1$. It can be shown that $\mathcal{A}$ is simple if and only if it satisfies the axiom

$$1; a; 1 = 1$$

for all non-zero $a \in \mathcal{A}$. Every relation algebra can be decomposed as a subdirect product of simple relation algebras (called the *components*) and the relation algebra is representable if and only if all the simple components are representable [JT48]. For a simple relation algebra, every representation is a disjoint union of square representations and thus a simple, representable relation algebra always has a square representation. **In this paper, unless otherwise stated, we assume that all relation algebras are simple and that all representations are square.**

Structural properties of relation algebras have been studied carefully, for example in [J82] and an overview of the theory of relation algebras can be found in [Mad91a, J91]. For a good history of the study of relation algebra try [Mad91b].

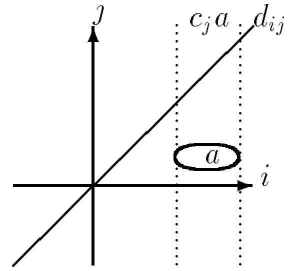
2.2 Cylindric Algebra

For higher order relations we use cylindric algebra. Let α be any ordinal — mostly finite, here. Corresponding to a proper relation algebra we define a cylindric set algebra of dimension α which is, roughly, a field of α-ary relations.

Definitions

- If U is a set and α an ordinal, ${}^{\alpha}U$ denotes the set of functions from α to U. A subset of ${}^{\alpha}U$ is called an α-ary relation on U. $D_{\kappa\lambda}$ denotes the set of all elements y of ${}^{\alpha}U$ such that $y(\kappa) = y(\lambda)$. Given an α-ary relation X on U define $C_\kappa X$ to be the set of all elements of ${}^{\alpha}U$ that agree with some element of X except, perhaps, on its κ'th co-ordinate.
- A *cylindric set algebra* of dimension α consists of a set S of α-ary relations on some domain U forming a field of sets and containing the diagonal elements $D_{\kappa\lambda}$ $(\kappa, \lambda < \alpha)$ and closed under the cylindrification operators C_κ $(\kappa < \alpha)$. The class of all cylindric set algebras of dimension α is denoted $\boldsymbol{Cs_\alpha}$.
- A *cylindric algebra* of dimension α is defined to be a structure $\mathcal{C} = (C, \vee, -, 0, 1, c_\kappa, d_{\kappa\lambda})_{\kappa,\lambda<\alpha}$ obeying the following axioms [HMT71] for every $x, y \in C, \kappa, \lambda, \mu < \alpha$:
  1. $(C, \vee, -, 0, 1)$ is a boolean algebra
  2. $c_\kappa 0 = 0$
  3. $x \leq c_\kappa x$
  4. $c_\kappa(x \wedge c_\kappa y) = c_\kappa x \wedge c_\kappa y$
  5. $c_\kappa c_\lambda x = c_\lambda c_\kappa x$
  6. $d_{\kappa\kappa} = 1$
  7. if $\kappa \neq \lambda, \mu$, then $d_{\lambda\mu} = c_\kappa(d_{\lambda\kappa} \wedge d_{\kappa\mu})$
  8. if $\kappa \neq \lambda$, then $c_\kappa(d_{\kappa\lambda} \wedge x) \wedge c_\kappa(d_{\kappa\lambda} \wedge -x) = 0$.
- A cylindric algebra is said to be *representable* if it is isomorphic to a subdirect product of cylindric set algebras. Such an isomorphism is called a *representation*. $\mathbf{RCA}_\alpha$ denotes the class of all representable cylindric algebras of dimension α. The reader is not required to know about subdirect decompositions to follow the rest of this paper.
- As with relation algebras, a cylindric algebra is *simple* if and only if it has no proper, non-trivial homomorphic images.
- A finite dimensional, simple cylindric algebra is representable if and only if it is isomorphic to a cylindric set algebra [HMT85].

The diagram below may help with the geometric interpretation of the operators.



The axioms, above, are valid over cylindric set algebras but, again, not every cylindric algebra is representable.

As with relation algebras, in a *cubic representation* X, we have $X(1) =^\alpha D$ where D is the domain. A *simple,* representable cylindric algebra always has a cubic representation.

2.3 Languages

For both relation and finite-dimensional cylindric algebra there is a very natural, first-order language corresponding to an algebra (see [McK66]). Let $\mathcal{A}$ be any relation algebra, and let $L = L(\mathcal{A})$ be the first-order language with one binary predicate symbol for each element of $\mathcal{A}$. We use the same symbol for an element of $\mathcal{A}$ as for the corresponding binary predicate in $L(\mathcal{A})$. This will not lead to ambiguity: for $a \in \mathcal{A}$, if we write $a(x, y)$, we are thinking of a as a relation symbol, but if we write simply a, we are thinking of a as an element of $\mathcal{A}$. Define an L-theory, $T_\mathcal{A}$ to consist of all of the following:

$$\begin{array}{rcll}
\sigma_{Id} & = & \forall x, y\,[Id(x,y) \leftrightarrow (x = y)] & : \text{ for each } R, S, T \in \mathcal{A} \text{ with} \\
\sigma_\vee(R,S,T) & = & \forall x, y\,[R(x,y) \leftrightarrow S(x,y) \vee T(x,y)] & : \; R = S \vee T \\
\sigma_\neg(R,S) & = & \forall x, y\,[1(x,y) \rightarrow (R(x,y) \leftrightarrow \neg S(x,y))] & : \; R = -S \\
\sigma_{\text{conv}}(R,S) & = & \forall x, y\,[R(x,y) \leftrightarrow S(y,x)] & : \; R = S^\smile \\
\sigma_;(R,S,T) & = & \forall x, y\,[R(x,y) \leftrightarrow \exists z(S(x,z) \wedge T(z,y))] & : \; R = S;T
\end{array}$$

It is clear that an L-structure is essentially a representation of $\mathcal{A}$ if and only if it is a model of $T_\mathcal{A}$.

Similarly if $\mathcal{C}$ is an n-dimensional cylindric algebra we can define a first-order language $L = L(\mathcal{C})$ with one n-ary predicate symbol for each element of $\mathcal{C}$. Again, if $a \in \mathcal{C}$ and we just write a we are thinking of it as an element of the cylindric algebra but, for any n-tuple of variables $x_0, \ldots, x_{n-1}$, the formula $a(x_0, \ldots, x_{n-1})$ treats the symbol a as an n-ary predicate of the language $L(\mathcal{C})$. As with relation algebra we can define an L-theory $T_\mathcal{C}$ in such a way that an L-structure is a representation of $\mathcal{C}$ if and only if it is a model of $T_\mathcal{C}$. As well as the boolean axioms for relation algebra $(\sigma_\vee(R,S,T) : R = S \vee T, \sigma_\neg(R,S) : R = -S,\ R,S,T \in \mathcal{C})$ there are two other axiom schemes:

$$\begin{array}{rcll}
\sigma_{ij} & = & \forall \bar{x}\,[d_{ij}(\bar{x}) \leftrightarrow (x_i = x_j)] & : \; i, j < n \\
\sigma_i & = & \forall \bar{x}\,[c_i a(\bar{x}) \leftrightarrow \exists y\, a(\bar{x}[i \rightarrow y])] & : \; i < n
\end{array}$$

where, here and throughout, $\bar{x}$ is taken to be an n-tuple of variables $(x_0, \ldots, x_{n-1})$ and $\bar{x}[i \rightarrow y]$ is the n-tuple obtained by replacing x_i by y in $\bar{x}$.

We can use these first-order languages to give an estimate of the expressive power of an algebraic logic. Before considering complexity, we need to define a *network.* This issue of complexity then arises when we consider the network satisfaction problem.

2.4 Networks

Networks are certain labeled, finite graphs — very widely used in computer science, notably in temporal reasoning [All84, DM87, VKvB89, DMP91, Hir96]. They are most easily defined for relation algebra, though the cylindric algebra case is exactly analogous.

Definitions Let $\mathcal{A}$ be any finite relation algebra.

- An $\mathcal{A}$-*network* $N = (D, f)$ (or just a network if the context is clear) is a finite set of nodes D and a function $f : D \times D \rightarrow \mathcal{A}$. Frequently, in the following, we use the same symbol N to denote the network, the nodes of the network and the labeling function, separating these different uses by the context. Thus $x \in N$ means that x is a node in the network N and $N(x, y)$ denotes the element of $\mathcal{A}$ labeling the edge (x, y) in the network.

- A network N is said to be *satisfiable* if there is a homomorphism h from N to some representation X of $\mathcal{A}$. So h maps the nodes of N to points in the representation in such a way that
$$(h(m), h(n)) \in X(N(m, n))$$
for any nodes $m, n \in N$.

- By the *network satisfaction problem* (NSP) we refer to the problem of deciding whether a given network is satisfiable in any representation, or not.

- A variant of this is the problem of deciding whether a network is satisfiable in a particular representation. If X is a representation we may refer to the *network satisfaction problem over X.*

- A *subnetwork* $N_1 = (D_1, f_1)$ of a network $N_2 = (D_2, f_2)$ is a network such that $D_1 \subseteq D_2$ and $f_2 \restriction_{D_1 \times D_1} = f_1$.

- An *atomic network* is a network where every edge is labeled by an *atom* of $\mathcal{A}$.

- A network N is said to be *transitively closed* if for all $x, y, z \in N$
$$\begin{aligned} N(x,x) &\leq Id \\ N(x,y) &= N(y,x)^{\smile} \text{ and} \\ N(x,z) &\leq N(x,y); N(y,z) \end{aligned}$$

- The *transitive closure* $TC(N)$ of a network N is transitively closed network with the same nodes as N, with $TC(N)(m, n) \leq N(m, n)$, for all $m, n \in N$ and with the labels on the edges 'as large as possible' subject to the previous conditions. It can be defined by iterating the following map until a fixpoint is found: $N \mapsto N^2$ where, for any edge (m, n)
$$N^2(m, n) \stackrel{def}{=} \bigwedge_{l \in N} N(m, l); N(l, n)$$
$TC(N)$ can be calculated in cubic time.

- A *zero network* N has every edge labelled by 0. If the transitive closure of N is a zero network then N is inconsistent. The converse does not hold, in general.

- Let N be a network. A *transitively closed labeling* L of N is a transitively closed atomic network with the same set of nodes as N such that for all edges e of L we have $L(e) \leq N(e)$. Given that the relation algebra is finite, the existence of a transitively closed labeling is a necessary, but not always sufficient, condition for the satisfiability of N (see the pentagonal algebra, section 3.2 for a case where not all transitively closed atomic networks are satisfiable. See [HH97a] for an analysis of what happens in the infinite case).

- Let k be a natural number. A network N is called *k-consistent* if for each subnetwork M of N with less than k nodes there is a transitively closed labeling of M.

For a finite, n-dimensional cylindric algebra $\mathcal{C}$ the definitions are quite similar, though the concept is less frequently used. A *cylindric $\mathcal{C}$-network* (or just network) N is a finite set of nodes D and a function $f :^n D \to \mathcal{C}$. As before we often refer to the nodes and the function by the same symbol N. A network N for an n-dimensional cylindric algebra $\mathcal{C}$ has labels on each n-tuple of nodes. When we wish to draw attention to this fact we may refer to N as an *n-dimensional $\mathcal{C}$-network*, or just an *n-dimensional network.* Accordingly, a network for a relation algebra, where edges are labeled, may be termed a *2-dimensional network*, though for the most part we will just refer to networks and determine their dimensions from the context.

The other definitions concerning networks are mostly unchanged for cylindric algebra.

Now for any relation algebra $\mathcal{A}$ we have defined the first-order language $L = L(\mathcal{A})$ with one binary predicate for each element of $\mathcal{A}$. A network N corresponds to a certain quantifier-free L-formula $\phi_N(\bar{x})$ where $\bar{x}$ is a sequence of variables with one variable x_n corresponding to each node n of N. Here is the definition:

$$\phi_N(\bar{x}) = \bigwedge_{m,n \in N} N(m,n)(x_m, x_n).$$

Recall that $N(m,n)(x,y)$ uses $N(m,n)$ as a binary predicate symbol from the language $L(\mathcal{A})$. Clearly N is satisfied in a representation X exactly when $X, h \models \phi_N(\bar{x})$ for some assignment h to the free variables. This is equivalent to $X \models \exists\bar{x}\phi_N(\bar{x})$. Thus, we can define a sublanguage L_{Net} of L consisting of all such formulas

$$L_{Net}(\mathcal{A}) = \{\phi_N(\bar{x}) : N \text{ is an } \mathcal{A}\text{-network } \}.$$

Similar definitions can be made for finite-dimensional cylindric algebras.

3 Complexity

We consider two satisfiability problems leading to two different complexity measures: the satisfaction problem and the network satisfaction problem. Given a finite, or at least recursively definable, relation algebra $\mathcal{A}$, the first problem, called the *satisfaction problem*, is to take an arbitrary $L(\mathcal{A})$-sentence ϕ and find out whether ϕ holds in any representation of $\mathcal{A}$. Equivalently the problem is to say whether $T_{\mathcal{A}} \cup \{\phi\}$ is consistent or not. The *time complexity* and *space complexity* of such problems are defined in any textbook on complexity e.g. [AHU74, vL94].

The network satisfaction problem (NSP) is a restriction of the satisfaction problem. Here, the problem is to decide whether an $L(\mathcal{A})$-sentence of the form $\exists\bar{x}\phi_N(\bar{x})$ (for some $\mathcal{A}$-network N) holds in some representation of $\mathcal{A}$. This is equivalent to asking whether N embeds in some representation of $\mathcal{A}$.

3.1 Complexity of the satisfaction problem

The complexity of the satisfaction problem is generally quite high.

THEOREM 1 *Let $\mathcal{A}$ be any representable relation algebra with $|\mathcal{A}| > 2$. The satisfaction problem for $\mathcal{A}$ is PSPACE-hard.*

PROOF:

We reduce the *quantified boolean formulas* (QBF) problem to the satisfaction problem for $\mathcal{A}$. QBF is known to be PSPACE complete [vEB94, page 41]. So, let ϕ be any quantified boolean formula using only propositional variables $p_0, \ldots, p_{k-1}$. We define a $L(\mathcal{A})$-formula ϕ^* in such a way that for any boolean valuation v there is a representation X of $\mathcal{A}$ and an assignment v^* to the free variables of ϕ^* such that $v(\phi) = \top$ if and only if $X, v^* \models \phi^*$. Moreover, it turns out that if ϕ^* is satisfiable in any representation of $\mathcal{A}$ then it is satisfiable in all representations of $\mathcal{A}$. ϕ^* has $k+1$ variables $w_\top, w_0, \ldots w_{k-1}$ and is obtained from ϕ by replacing every quantifier $\exists p_i$ by $\exists w_i$ (and $\forall p_i$ by $\forall w_i$) and all instances of p_i are replaced by $Id(w_i, w_\top)$ $(i < k)$. So, for example if $\phi = \forall p_0 \exists p_1 (p_0 \to \neg p_1)$ then $\phi^* = \forall w_0 \exists w_1 \, (Id(w_0, w_\top) \to \neg Id(w_1, w_\top))$.

Since $|\mathcal{A}| > 2$ it follows that any representation X of $\mathcal{A}$ has $|X| > 1$. Pick any two distinct points $x_\top, x_- \in X$. Given a boolean valuation v we can construct an assignment to the variables v^* by letting $v^*(w_\top) = x_\top$, $v^*(w_i) = x_\top$ if $v(p_i) = \top$ and $v^*(w_i) = x_-$ if $v(p_i) = -$. A simple induction on ϕ shows that $v(\phi) = \top$ if and only if $X, v^* \models \phi^*$. On the other hand any assignment w^* to the free variables of ϕ^* making $X, w^* \models \phi^*$ gives rise to a valuation w making ϕ true, namely $w(p_i) = \top$ if and only if $w^*(p_i) = w^*(w_\top)$. Thus we have a reduction of QBF to the satisfaction problem for $\mathcal{A}$ which proves the result.

□

The proof that the satisfaction problem is always PSPACE hard was fairly trivial. For many relation algebras it may be possible to prove a higher complexity or even that the satisfaction problem is undecidable.

PROBLEM 1 *Find a relation algebra $\mathcal{A}$ such that the satisfaction problem is undecidable over $\mathcal{A}$. Or, even better, find $\mathcal{A}$ such that the network satisfaction problem is undecidable over $\mathcal{A}$.*

3.2 Complexity of the network satisfaction problem for various relation algebras

Here we introduce some well-known relation algebras: the point algebra, the Allen interval algebra, the left-linear algebra, the containment algebra and the metric point algebra of [DMP91] and others. Each of these has had wide application in temporal reasoning, databases and planning (e.g. [AK83b, DM87, Pel88, Hir96, Hir95] etc.). We give the complexity of the network satisfaction problem for each case if it is known and refer to a new result for the complexity of the NSP for the left-linear relation algebra, the proof of which is deferred to a later section.

Before giving the examples it should be noted that all but one of the relation algebras, below, are finite and all are atomic. The axioms for relation algebras include a distribution rule but in fact it is possible to derive from these axioms an infinite distribution rule

$$a;(\bigvee_\lambda b_\lambda) = \bigvee_\lambda (a;b_\lambda)$$

and a similar rule for distribution from the right and for distribution of converse over arbitrary disjunctions. It follows that the composition table for an atomic relation algebra is determined by the compositions of the atoms. The converse of an arbitrary element can also be calculated if the converses of all the atoms are known. Using this, an element of an atomic relation algebra may be identified uniquely by the set of all atoms beneath it:

$$a \mapsto \{\alpha \in At(\mathcal{A}) : \alpha \leq a\}$$

In this view each element of an atomic relation algebra is a set of atoms. Thus if $a \in \mathcal{A}$ and $\alpha \in At(\mathcal{A})$ we will write $\alpha \in a$ rather than $\alpha \leq a$.

Each non-atomic element will now be denoted by the set of atoms beneath it. The atoms are now just singleton sets $\{\alpha\}$, but we will simply write α for an atomic element like this.

$\mathcal{A}_2$ The simplest possible, non-degenerate relation algebra has two atoms, $Id, \#$, and consists of all four sets of atoms. Both atoms are self-converse and composition is defined by the table below.

;	Id	#
Id	Id	#
#	#	1

A square representation X of $\mathcal{A}_2$ can be obtained by taking for its domain any set D with $|D| > 2$ and $X(Id) = \{(d,d) : d \in D\}$, $X(\#) = \{(d_1, d_2) \in D : d_1 \neq d_2\}$; indeed any square representation X of $\mathcal{A}_2$ arises in this way. Every transitively closed, non-zero $\mathcal{A}_2$-network N is satisfied in some square representation of $\mathcal{A}_2$ — for example take any square representation R of size greater than or equal to $|N|$. For a homomorphism from N to the representation, take any map $f : N \to R$ such that if $m, n \in N$ and $\# \in N(m,n)$ then $f(m) \neq f(n)$, and if $N(m,n) = Id$ then $f(m) = f(n)$. Thus, the network satisfaction problem has cubic complexity as the transitive closure of a network can be calculated in cubic time.

$\mathcal{A}_2'$ This relation algebra is very similar to $\mathcal{A}_2$ but the composition table is

;	Id	$\#$
Id	Id	$\#$
$\#$	$\#$	Id

This has a representation X consisting of just two points a, b. Here $X(\#) = \{(a,b),(b,a)\}$. All representations of $\mathcal{A}_2'$ are isomorphic to X. Any transitively closed network is satisfiable in X and so the network satisfaction problem has, at worst, cubic complexity.

$\mathcal{A}_3$ This graph coding relation algebra has three self-converse atoms Id, e, d (e is intended to mean there is a graph edge and d means that there isn't). Composition is defined by the table

;	e	d
e	1	$\{e,d\}$
d	$\{e,d\}$	1

(we omit the entries for Id). A graph G defines a representation of $\mathcal{A}_3$ if for every pair of nodes $g_1, g_2 \in G$ there is a node connected to g_1 and g_2; there is a node connected to neither g_1 nor g_2; and there is a node connected to g_1 but not g_2. Every transitively closed $\mathcal{A}_3$-network is satisfied in the representation of $\mathcal{A}_3$ defined by the *random graph.* The complexity of the NSP is therefore, at worst, cubic.

The Point Algebra $\mathcal{P}$ has three atoms: $Id, <$ and $>$. $1 = \{Id, <, >\}$, the identity is Id (self-converse) and the converse of $<$ is $>$. Composition is defined by the table below.

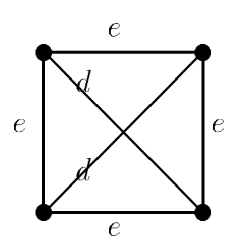
;	Id	$<$	$>$
Id	Id	$<$	$>$
$<$	$<$	$<$	1
$>$	$>$	1	$>$

Any square representation of $\mathcal{P}$ must be a dense linear order without endpoints and so the rational numbers with their usual ordering embed in any representation of $\mathcal{P}$ and any countable, square representation must be base-isomorphic to the rationals. So the general NSP and the NSP over X (any given square representation X) are equivalent. It turns out that any transitively closed network is satisfiable in the rationals and so the network satisfaction problem for $\mathcal{P}$ has cubic complexity (see [VK86] with an important correction in [VKvB89]).

The Pentagonal Algebra $\mathcal{PA}$ has three atoms Id, e, d all self-converse, composition is defined by

;	e	d
e	$\{Id, d\}$	$\{e, d\}$
d	$\{e, d\}$	$\{Id, e\}$

It is not hard to check that $\mathcal{PA}$ has exactly one representation X, up to isomorphism. The domain of X has five points $0, \ldots, 4$. $X(e) = \{(i,j) : |i-j| = 1 (\text{mod } 5)\}$ and $X(d) = \{(i,j) : |i-j| = 2 (\text{mod } 5)\}$. Here is an example of a relation algebra where a transitively closed atomic network is not always satisfiable, as illustrated by the network below[2].



[2]This example is based on a discussion between Roger Maddux and the author.

It is rather irritating for a relation algebra with only one, finite representation, but we have no better estimate of the complexity of the network satisfaction problem over $\mathcal{PA}$ except to say that it lies in **NP**.

For more about relation algebras with at most three atoms read [AM94] where the interested reader can find the smallest representations.

The Allen interval algebra This algebra ($\mathcal{I}$) has thirteen atoms: Id, precedes, meets, overlaps, starts, during, ends together with the converses of the last six. The composition table can be found in [All83]. A natural representation of $\mathcal{I}$ is obtained by taking as domain all ordered pairs of rational numbers (p, q) with $p < q$. Each of the thirteen atoms is then interpreted in the obvious way, for example (p, q) meets (r, s) if and only if $q = r$. It was proved in [LM94] that all countable, square representations of $\mathcal{I}$ are base-isomorphic to the one just outlined and so, again, the general NSP and the NSP over some fixed representation X are equivalent. It turns out that any transitively closed atomic network is satisfiable (hence the NSP has cubic complexity for *atomic* networks) but for general networks transitive closure does not guarantee satisfiability [All84] and the NSP has been shown to be an NP complete problem [VK86].

The Left Linear Point Algebra The left linear point algebra was first presented in [Com83], where it is referred to as $\mathcal{N}_1$. A concrete representation of it appeared in [D̈91], see also [AGN94], page 642.

A *left-linear* structure $(L, <)$ is a partial order such that if $s, t, u \in L$ and $s, t < u$ then either $s < t, t < s$ or $s = t$. The algebraic counterpart to this type of structure is the relation algebra $\mathcal{L}$ which has four atoms: $Id, <, >, \sharp$ and the composition table is:

;	$<$	$>$	$\sharp$
$<$	$<$	$-\sharp$	$\{<, \sharp\}$
$>$	1	$>$	$\sharp$
$\sharp$	$\sharp$	$\{>, \sharp\}$	1

A representation of this is more difficult to define but it is determined by a dense partial order, left linear, densely branching and without any endpoints (see section 4.1). This relation algebra should be useful for modeling flows of time which branch into the future, but where the past is fixed.

We will show later (theorem 9) that although a non-zero transitive closure does not guarantee that an $\mathcal{L}$-network is satisfiable, there is a p-time algorithm to test consistency.

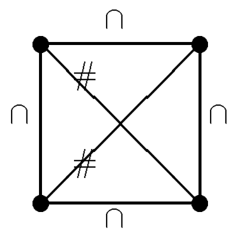
The Containment Algebra The Allen interval algebra has a subalgebra $\mathcal{C}$ with five atoms:

$$\begin{array}{rcl} Id & & \\ \text{`contained-in'} & = & \{\text{ starts, during, ends}\} \\ \text{`contains'} & = & \{\text{starts}^{\smile}, \text{ during}^{\smile}, \text{ ends}^{\smile}\} \\ \cap = \text{`intersects'} & = & \{\text{meets, overlaps, meets}^{\smile}, \text{ overlaps}^{\smile}\} \\ \# = \text{`disjoint'} & = & \{\text{precedes, precedes}^{\smile}\}. \end{array}$$

One way of finding a representation of $\mathcal{C}$ is to take any representation of $\mathcal{I}$ and then take the restriction to $\mathcal{C}$. Thus, two intervals are related by the atom 'contained-in' if and only if they are related by { starts, during, ends}.

A more general way of building a representation of $\mathcal{C}$ is to take any uncountable set S and define an equivalence relation $\sim$ on the power set of S where $U \sim V$ if the symmetric difference of U and V is finite. Then $\mathcal{P}(S)/\sim$ is an atomless boolean algebra. Define a representation X of $\mathcal{C}$ by letting the domain D consist of all equivalence classes of countably infinite subsets of S. The elements of $\mathcal{C}$ can be represented in the natural way as binary relations over D, for example $[U]$ 'contains' $[V]$ if all but finitely many elements of U are members of V and there are infinitely many elements of $U \setminus V$. [Here $[U]$ is the equivalence class of X under $\sim$.] Similarly, $[U]$ 'intersects' $[V]$ means that the following sets are all infinite: $U \cap V; U \setminus V; V \setminus U$.

Here we have an example where the network satisfaction problem and the NSP over X can be different. There is an example of an atomic network (below) taken from [LM88, page 41] which does not embed in any representation of the Allen interval algebra but it does embed in the representation based on the atomless boolean algebra.



Neither the complexity of the network satisfaction problem for $\mathcal{C}$ nor the network satisfaction problem for $\mathcal{C}$ over representations of $\mathcal{I}$ is known to us.

The Metric Point Algebra This relation algebra $\mathcal{M}$ was first defined in [DMP91]. The elements of $\mathcal{M}$ are all finite unions of intervals with rational endpoints together with unbounded intervals e.g. $[2, 3) \cup (7, 9\frac{1}{2}) \cup [11, \infty)$. Open, closed and semi-open intervals are included. The identity element is $[0, 0]$; the top element is $(-\infty, \infty)$; the converse of $[p, q]$ is $[-q, -p]$ (with similar definitions for open and semi-open intervals); negation is defined by $-[p, q] = (-\infty, p) \cup (q, \infty)$ and composition is defined by

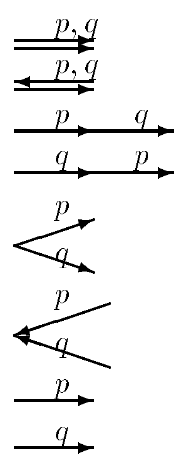
$$[p, q]; [r, s] = [p + r, q + s].$$

A representation Ξ of $\mathcal{M}$ can be obtained by taking as the domain the rational numbers and letting $\Xi([p, q]) = \{(r, s) \in \mathbb{Q} : s - r \in [p, q]\}$.

For $\mathcal{M}$ the NSP is NP-complete, but if we restrict to networks where each edge is labeled by a *single* interval then consistency can be checked in cubic time [DMP91].

The Pairs Algebra $\mathcal{B}_7$ Consider the following, seven-atom proper relation algebra $\mathcal{B}(S)$. The domain consists of distinct pairs from any set S of size at least six. Using only equality and inequality, there are seven ways (shown below) that two pairs p, q can relate:

	Id
$sw =$	Swap
$m =$	Meets
$m^{\smile} =$	Met-by
$st =$	Starts
$e =$	Ends
$\sharp =$	Disjoint



The converse of 'Meets' is 'Met-by' and all other atoms are self-converse. Composition is given by

;	sw	m	$m^\smile$	st	e	$\sharp$
sw	Id	st	e	m	$m^\smile$	$\sharp$
m	e	$\{m^\smile, \sharp\}$	$\{Id, e\}$	$\{sw, m\}$	$\{st, \sharp\}$	$\{st, m^\smile, \sharp\}$
$m^\smile$	st	$\{Id, st\}$	$\{m, \sharp\}$	$\{e, \sharp\}$	$\{m^\smile, sw\}$	$\{m, e, \sharp\}$
st	$m^\smile$	$\{e, \sharp\}$	$\{m^\smile, sw\}$	$\{Id, st\}$	$\{m, \sharp\}$	$\{m, e, \sharp\}$
e	m	$\{m, sw\}$	$\{st, \sharp\}$	$\{m^\smile, \sharp\}$	$\{Id, e\}$	$\{m^\smile, st, \sharp\}$
$\sharp$	$\sharp$	$\{m^\smile, e, \sharp\}$	$\{m, st, \sharp\}$	$\{m^\smile, e, \sharp\}$	$\{m, st, \sharp\}$	1

The composition table does not depend on S, provided $|S| \geq 6$. Let us define $\mathcal{B}_7$ to be the (abstract) relation algebra isomorphic to $\mathcal{B}(S)$ (any S with $|S| \geq 6$). We'll show later (corollary 27) that the NSP for $\mathcal{B}_7$ is NP-complete.

In fact every square representation of $\mathcal{B}_7$ is base-isomorphic to a proper relation algebra $\mathcal{B}(S)$ (for some set S) i.e. a representation where the domain consists of pairs of points.

THEOREM 2 *Let X be a square representation of $\mathcal{B}_7$. There is a set S and a base-isomorphism from X onto $\mathcal{B}(S)$.*

PROOF:

Let X be any square representation of $\mathcal{B}_7$. Let X_0, X_1 be disjoint, base-isomorphic copies of X. For each $x \in X$, let $x_0 \in X_0, x_1 \in X_1$ be the elements of the copy representations corresponding to x. Let $D = X_0 \cup X_1$ and let $\sim$ be the smallest binary relation on D such that for all $x, y \in X$:

- $\sim$ is an equivalence relation
- $(x, y) \in X(\{st, Id\}) \Rightarrow x_0 \sim y_0$
- $(x, y) \in X(\{m^\smile, sw\}) \Rightarrow x_0 \sim y_1$
- $(x, y) \in X(\{m, sw\}) \Rightarrow x_1 \sim y_0$
- $(x, y) \in X(\{Id, e\}) \Rightarrow x_1 \sim y_1$

Now let $S = D/\sim$, let $[d]$ be the equivalence class of $d \in D$. The mapping where, for all $x \in X$, $x \mapsto ([x_0], [x_1])$ is a base-isomorphism from X to a representation of $\mathcal{B}_7$ to a domain consisting of all pairs from S. $\Box$

COROLLARY 3 *$\mathcal{B}_7$ is weakly ω-categorical i.e. all its countably infinite, square representations are base-isomorphic to $\mathcal{B}_7(\omega)$.*

COROLLARY 4 *A $\mathcal{B}_7$-network N is satisfiable if and only if it is satisfiable in $\mathcal{B}_7(\omega)$.*

PROOF:

Let N be satisfied in some representation X. By the theorem, X is base-isomorphic to $\mathcal{B}(S)$, for some set S. So let h embed N in $\mathcal{B}(S)$. Now let ι be any injection from S into ω. Then $h \circ \iota$ is an embedding of N in the representation $\mathcal{B}_7(\omega)$. $\Box$

3.3 Complexity of the network satisfaction problem for one cylindric algebra

Here we define the simplest possible four-dimensional cylindric algebra $\mathcal{D}_4$. We'll show how to reduce the Hamiltonian circuit problem to the network satisfaction problem over $\mathcal{D}_4$ and thus that the latter problem is NP-hard. Later, when we develop the machinery of interpretations, we will be able to use this result to show that most cylindric algebras of dimension four or more have an NP-hard NSP. The results obtained here are similar to results found in [Hir94b] where a certain type of relation algebra (called a *pair algebra*) is shown to have NP-hard NSP.

Recall that, for $\mathcal{C} \in \mathbf{CA_4}$, a $\mathcal{C}$-network N is a set of nodes, N_1, and a map $N_2 : (N_1)^4 \to \mathcal{C}$.

Definition of $\mathcal{D}_4$ $\mathcal{D}_4$ is the simplest possible non-degenerate four-dimensional cylindric algebra, being the minimal algebra generated by the diagonals. It has fifteen atoms listed below, the indices i, j, k, l range over $0, 1, 2, 3$ and *all indices are constrained to be distinct.*

eq	$=$	$\bigwedge_{i,j} d_{ij}$	all four equal
δ_k	$=$	$(\bigwedge_{i,j\neq k} d_{ij}) \wedge \bigwedge_i -d_{ik}$	only kth element different
$eq_{\{ij\}}$	$=$	$d_{ij} \wedge \bigwedge_{k,l}(-d_{ik} \wedge -d_{jk} \wedge -d_{kl})$	only i'th and j'th are equal
$eq_{\{\{ij\},\{kl\}\}}$	$=$	$d_{kl} \wedge d_{ij} \wedge -d_{ki} \wedge -d_{kj} \wedge -d_{li} \wedge -d_{lj}$	(ith element $=$ jth) $\neq$ (kth $=$ lth)
$diff$	$=$	$\bigwedge_{i,j} -d_{ij}$	all different

The number of atoms of each kind is 1, 4, 6, 3 and 1. Note that every atom is an intersection of diagonals and negated diagonals so every element of $\mathcal{D}_4$ is a boolean combination of diagonals.

A representation X of $\mathcal{D}_4$ is obtained by taking any set S with $|S| \geq 5$ and mapping the atoms of $\mathcal{D}_4$ to quartic relations over S in the natural way. For example

$$X(\delta_3) = \{(s_0, s_1, s_2, s_3) \in S : s_0 = s_1 = s_2 \neq s_3\}$$

Let X_ω be the cubic representation obtained in this way by letting $S = \omega$.

LEMMA 5 *Any $L(\mathcal{D}_4)$-network N is satisfiable in some representation if and only if it is satisfiable in X_ω.*

PROOF:

Let h be a homomorphism from N into some representation X of $\mathcal{D}_4$. Let π be any injection from $h(N)$ $(= \{h(n) : n \in N\} \subseteq X)$ into X_ω. Then $h \circ \pi$ is a homomorphism of N into X_ω and so N is satisfied in X_ω.
□

The elements of $\mathcal{D}_4$ become quartic relations in a representation, but we can also think of them as binary relations on pairs. For any representation X of $\mathcal{D}_4$, let us call a pair of points $(x_1, x_2) \in X$ an *interval* if $x_1 \neq x_2$. Below we identify and name certain elements of $\mathcal{D}_4$ that relate two intervals (x_0, x_1) and (x_2, x_3).

$$\begin{aligned} Id &= eq_{\{\{02\},\{13\}\}} \\ same &= eq_{\{\{02\},\{13\}\}} \vee eq_{\{\{03\},\{12\}\}} \\ starts &= eq_{\{02\}} \\ ends &= eq_{\{13\}} \\ meets &= eq_{\{12\}} \end{aligned}$$

As with relation algebra we will use the notation whereby a non-atomic element of $\mathcal{D}_4$ is written as the set of atoms beneath it. Thus we will write $same = \{eq_{\{\{02\},\{13\}\}}, eq_{\{\{03\},\{12\}\}}\}$. In any representation $same$ holds on (x^-, x^+, y^-, y^+) if $(x^-, x^+), (y^-, y^+)$ are both intervals and $\{x^-, x^+\} = \{y^-, y^+\}$, but note that their orders may be reversed. The meanings of the other elements can be gathered from their names.

THEOREM 6 *The network satisfaction problem is NP-hard over $\mathcal{D}_4$.*

PROOF:

The proof works be reducing the Hamiltonian circuit problem to the NSP over $\mathcal{D}_4$. Let G be any undirected graph with n nodes. We construct a $\mathcal{D}_4$-network $N(G)$, in time polynomial in n, such that G has a Hamiltonian circuit if and only if $N(G)$ is satisfiable in some representation of $\mathcal{D}_4$.

1. Make a directed graph G' with the same nodes as G by arbitrarily choosing a direction for each edge of G.

2. Make a $\mathcal{D}_4$-network M having two distinct nodes e^-, e^+ for each edge e of G'. $|M| \leq 2n^2$. Let e, f be edges from G'. Set $M(e^-, e^+, f^-, f^+) = diff$ if the two edges e and f are disjoint in G'; $M(e^-, e^+, f^-, f^+) = meets$ if e meets f in G' (i.e. the end of e coincides with the start of f); $M(e^-, e^+, f^-, f^+) = meets^{\smile}$ if f meets e; $M(e^-, e^+, f^-, f^+) = starts$ if the edges are not equal but start together in G' and $M(e^-, e^+, f^-, f^+) = ends$ if the edges e and f are not equal but share the same endpoint. Let $M(e^-, e^+, e^-, e^+) = Id$ for all edges $e \in G'$ and let any four-tuple not covered by the above be labelled by 1. Note that
   - M is consistent. For this, make a representation of $\mathcal{D}_4$ by taking for the domain the nodes of G' plus extra points, if necessary, so that the size of the representation is at least 5. Then, to map M into the representation, for each edge $e \in G'$ map e^-, e^+ to the two nodes of the edge e in order. We'll refer to this construction as the model for M *based on* G'.
   - If h is a homomorphism from M into some representation X of $\mathcal{D}_4$ then $\{((h(e^-), h(e^+)) : e \in G'\}$ forms a graph isomorphic to G'.
   - For distinct edges e and f from G' the relation $M(e^-, e^+, f^-, f^+)$ is disjoint from $same$.
3. Extend M to M^+ by adding n distinct, new intervals $f_0, \ldots, f_{n-1}$ (with $f_i = (f_i^-, f_i^+)$) in such a way that
   - for each i, if h is any homomorphism from M^+ to a representation X, there must be some edge $e \in G'$ with $(h(f_i^-), h(f_i^+), h(e^-), h(e^+)) \in X(same)$
   - for each edge $e \in G'$ and each $i < n$, there is a homomorphism h from M^+ to some representation X with $(h(f_i^-), h(f_i^+), h(e^-), h(e^+)) \in X(same)$. Indeed, further, for any sequence of edges $e_0, \ldots, e_{n-1}$ from G' there is a homomorphism h from M^+ to some representation X with $(h(f_i^-), h(f_i^+), h(e_i^-), h(e_i^+)) \in X(same)$ (all $i < n$).

   This construction is given later.
4. Define $N(G)$ with the same nodes as M^+ by $N(G) \restriction_M = M$ (the labels are unchanged on the old nodes) and

$$N(G)(f_i, f_{i+1}) = meets$$

for $i = 0 \ldots n-2$,

$$N(G)(f_{n-1}, f_0) = meets$$

and for all the other new intervals f_i, f_j with $|i - j| \neq 0, 1 (\text{mod } n)$

$$N(G)(f_i, f_j) = diff.$$

If G does contain a Hamiltonian circuit then a model of $N(G)$ can be obtained, based on G, by letting the f_i be the edges of a Hamiltonian circuit. $N(G)$ is therefore consistent. Conversely, if $N(G)$ is consistent then in any model, the definition of $N(G)$, part 4, enforces that the intervals f_i form a Hamiltonian circuit on a graph isomorphic to G.

It remains to show how to perform the construction in part 3. Let N be any consistent $\mathcal{D}_4$-network and $S \subseteq N \times N$ be any set of intervals such that for distinct intervals $s \neq t \in S$ we have $N(s, t) \wedge same = 0$. We show how to extend N to N^+ so that N^+ includes an interval $f = (f^-, f^+)$ and for any homomorphism h from N^+ to a representation X, we have $(h(f^-), h(f^+), h(s^-), h(s^+)) \in X(same)$ for some interval $s = (s^-, s^+) \in S$. (Or $h(\{f^-, f^+\}) = h(\{s^-, s^+\})$ in X).

Also we show that for any $s \in S$ there is a homomorphism g from N^+ to some representation Y such that $(g(f^-), g(f^+), g(s^-), g(s^+)) \in Y(same)$. The size of this extension will be bound by a polynomial in $|N|$.

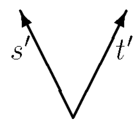
First group the intervals in S in pairs (possibly with an odd one left). For each such pair $(s = (s^-, s^+), t = (t^-, t^+)) \in S$ $(s \neq t)$ we add the nodes s', t', w and f_{st} and set

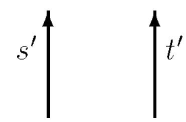
$$N^+(s', s) = N^+(t', t) = same,$$

and

$$N^+(s', t') = \{diff, starts\}.$$

In any model of N^+, this constrains s', t' to lie on the same edges as s and t (respectively) though possibly in the opposite directions, and s', t' look like one of the two diagrams below.





This is where we use the assumption that $N(s, t) \wedge same = 0$. Now let

$$N^+(w, s') = N^+(w, t') = \{eq_{\{03\}}, eq_{\{13\}}\}$$

so w must join the two 'top ends' of s' and t'. Finally let

$$N^+(f_{st}, w) = \{eq_{12}, eq_{13}\}$$

$$N^+(f_{st}, s') = N^+(f_{st}, t') = \{Id, starts, diff\}.$$

f_{st} finishes at one or the other endpoint of w (so it can't be disjoint from both s' and t') and the second constraint forces f_{st} to be equal to either s' or t'. For any homomorphism h mapping N into a representation X it is possible to extend h to a homomorphism h^+ of N^+ into X so that $(h^+(f_{st}) = h^+(s')$ but it is also possible to choose the extension h^+ so that $h^+(f_{st}) = h^+(t')$.

Also, for any homomorphism g from N^+ into a representation Y, the interval f_{st} must map under g to the same interval as one or other of s and t, though possibly in the reverse direction.

We now have a set of new nodes of the form f_{st}, about half as many as we started with, and if $\{s, t\} \cap \{u, v\} = \emptyset$ then f_{st} and f_{uv} still share at most one endpoint in any model of N. Therefore we can repeat the whole procedure and construct new nodes f_{stuv} that must coincide with one of f_{st} or f_{uv} i.e. they coincide with one of s, t, u or v. This process is repeated about $\log(n)$ times until there is a single node f. Any homomorphism of N^+ makes f the same as one of the intervals in S (though perhaps in the reverse order) and for each interval $s \in S$ there exists a homomorphism of N^+ making f be the same as s.

Returning to the construction of $N(G)$, part 3, this is done for each of the intervals f_i. Each interval f_i can consistently be the same as any edge of G' but must always be the same as some edge of G', in a model based on G', and because there is no constraint imposed between the f_is it is possible to assign the intervals $f_0, \ldots, f_{n-1}$ to any sequence of edges from G'.

If the original graph G has n nodes then M has no more than $2n^2$ nodes (two for each edge of G'). One iteration of the construction of M^+ adds on no more than $n^2 \times 4$ extra nodes and the total number of nodes added in the construction of each f_i is bound by $4n^2 \sum_{j=0}^{logn} (\frac{1}{2})^j < 8n^2$. Thus M^+ has no more than $2n^2 + n \times 8n^2 \approx 8n^3$ nodes — certainly bound by a polynomial in n.

□

4 Homogeneous Representations

The concept of homogeneity is relevant to our discussion of expressive power because, as we'll see shortly, any first-order formula is equivalent to a quantifier-free formula over this type of representation. Put roughly, in a homogeneous representation the *context* of any finite substructure is always the same. Thus, for example, the network satisfaction problem becomes context free and this may reduce the complexity of the problem.

Definition

- A *local isomorphism* of a relational structure X is a finite map $\theta : \bar{x} \to X$ preserving all the relations. If X is a representation of a relation algebra $\mathcal{A}$, where we have only binary predicates, this means that for all $a \in \mathcal{A}$, for all $x_i, x_j \in \bar{x}$

$$(x_i, x_j) \in X(a) \Leftrightarrow (x_i\theta, x_j\theta) \in X(a)$$

  Since $Id \in \mathcal{A}$ this forces a local isomorphism to be injective.

- A *base-automorphism* of a relational structure X is a permutation of the domain of X, preserving all the relations — i.e. a base-isomorphism from X onto itself. The base-automorphisms of a relational structure form a group.

- X is said to be *n-transitive*[3] if any local isomorphism of size n or less extends to a base-automorphism of X.

- If X is n-transitive for all $n \in \mathbb{N}$ we call X *homogeneous.*

It is not hard to check that for any formula $\phi(\bar{x})$, any base-automorphism ρ of X and any tuple of elements $\bar{a} \in X$,

$$X \models \phi(\bar{a}) \Leftrightarrow X \models \phi(\rho(\bar{a})).$$

4.1 Examples of homogeneous representations

All relation algebras mentioned here are defined in section 3.2.

- Any square representation of $\mathcal{A}_2$ is homogeneous. To see this, observe that a local isomorphism of a square representation is any finite, one-one map. Such a map can always be extended to a permutation of the domain, which is a base-automorphism of the representation.

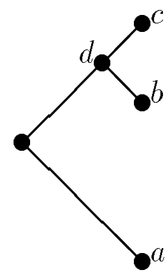
- The representation of $\mathcal{P}$ based on the rationals $\mathbb{Q}$ is, perhaps, the classic case of a homogeneous representation. For this, note that a local isomorphism is any finite, order preserving map from $\mathbb{Q}$ to $\mathbb{Q}$. It is always possible, using a back and forth construction, to extend such a map to a full base-automorphism of $\mathbb{Q}$.

- Another standard example of a homogeneous representation is the *random graph*, considered as a representation of $\mathcal{A}_3$.

- The representation of $\mathcal{I}$ based on ordered pairs of rational numbers is also homogeneous. We'll prove this later using the idea of an *interpretation* (lemma 17).

- The left-linear relation algebra $\mathcal{L}$ has no homogeneous representation. For this observe first, that it follows from the definition of composition in $\mathcal{L}$ and the definition of a representation, that a network with nodes a, b, c, d with $d < c$, $d < b$, $d \sharp a$, $c \sharp b$ (below) must embed in any representation of $\mathcal{L}$.

[3] The integral relation algebras possessing 1-transitive representations are called *permutational relation algebras* in [McK66].

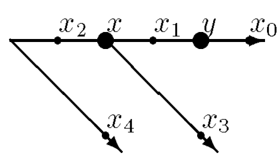


Now the map $\theta : a \mapsto b,\ b \mapsto a,\ c \mapsto c$ is a local isomorphism as each distinct pair from a, b, c are related by $\#$. However θ cannot extend to an automorphism of the representation as d would have to map to a point less than c and a but not less than b. This is impossible. Thus, a representation of $\mathcal{L}$ can never be 3-transitive, and hence can't be homogeneous.

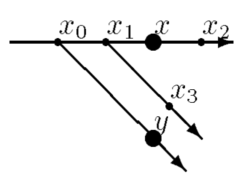
However $\mathcal{L}$ does have a 2-transitive, countable representation X. A construction of this representation can be found in [AGN94, pages 640–642] together with a reference to an earlier construction [D$\ddot{9}$1], pages 12–13. Formally, the domain of the representation X is $\mathbb{Q}^* = \bigcup_{n \in (\mathbb{N} \setminus 0)} ({}^n\mathbb{Q})$. Let $f \in {}^{n+1}\mathbb{Q}$ and $g \in {}^{m+1}\mathbb{Q}$. The pair $(f, g) \in X(<)$ if and only if

1. $n \leq m$ and $f \restriction_n = g \restriction_n$ and
2. if $n < m$, then $f(n) \leq g(n)$; if $m = n$, then $f(n) < g(n)$.

There are two important properties of this representation that follow from the composition table for $\mathcal{L}$. Firstly let $(x, y) \in X(<)$. x and y are not endpoints and there is a branch in between them. So there are points x_0, x_1, x_2, x_3, x_4 as in the diagram.



Secondly, suppose $(x, y) \in X(\sharp)$. This time we can find points x_0, x_1, x_2, x_3 as bellow.



Now any countable representation X is 2-transitive for if $d_1, e_1, d_2, e_2 \in S$ with $d_1 < e_1$ and $d_2 < e_2$ then the map $d_1 \mapsto d_2,\ e_1 \mapsto e_2$ extends to a full base-automorphism of the representation. The base-automorphism is obtained by a back and forth construction and uses the fact that the representation is countable, densely branching without endpoints. If $d_1 \# e_1$ and $d_2 \# e_2$ then a base-automorphism extending the map can be found in the same way.

The next theorem shows that over homogeneous representations the quantifier-free formulas are fully expressive. First a definition.

Definition Let A be a structure for a language $\mathcal{L}$. i.e. A is a domain with an interpretation of the constants, functions and relations of $\mathcal{L}$. Two formulas $\phi(\bar{x})$ and $\psi(\bar{x})$ with the same free variables $\bar{x}$ are called *equivalent* over A if for any assignment h to the free variables $\bar{x}$ we have

$$A, h \models \phi(\bar{x}) \Leftrightarrow A, h \models \psi(\bar{x})$$

THEOREM 7

1. *Let X be a homogeneous representation of $\mathcal{A}$. Let $\phi(\bar{x})$ be any formula of the infinitary language $L_{\infty\omega}(\mathcal{A})$, where infinite disjuncts and conjuncts are allowed, but only finitely many variables. $\phi(\bar{x})$ is equivalent, over X, to a quantifier-free formula $\psi(\bar{x}) \in L_{\infty\omega}(\mathcal{A})$.*
2. *If X is a homogeneous representation of a finite relation algebra $\mathcal{A}$ and $\phi(\bar{x})$ is any $L(\mathcal{A})$-formula (ordinary first-order formula), then $\phi(\bar{x})$ is equivalent, over X, to a quantifier-free $L(\mathcal{A})$-formula.*

PROOF:

1. The proof works by induction on the construction of formulas. The only problematic case is the one which deals with quantification. So let $\phi(\bar{x}) = \exists y\ \theta(y, \bar{x}) \in L_{\infty\omega}(\mathcal{A})$ be any formula such that $\theta(y, \bar{x})$ is quantifier free.

   Definitions

   - Let $\Theta(\bar{x})$ be a set of quantifier-free $L_{\infty\omega}(\mathcal{A})$-formulas using only variables from $\bar{x}$. We may omit reference to the variables $\bar{x}$ if it is not needed. Θ is *downward closed* if
     - $(\bigvee_{\lambda\in\Lambda} \phi_\lambda) \in \Theta$ implies that for some $\lambda \in \Lambda$, $\phi_\lambda \in \Theta$.
     - $(\bigwedge_{\lambda\in\Lambda} \phi_\lambda) \in \Theta$ implies that for all $\lambda \in \Lambda$, $\phi_\lambda \in \Theta$.
     - for $a \in \mathcal{A}$, $\neg(a(x, y)) \in \Theta$ implies that $(-a)(x, y) \in \Theta$.

     The *downward closure* $DC(\Phi(\bar{x}))$ of a set $\Phi(\bar{x})$ of quantifier-free formulas is the smallest downward closed set containing $\Phi(\bar{x})$.
   - Θ is *consistent* if
     - for $a \in \mathcal{A}$ and any variable x, $a(x, x) \in DC(\Theta)$ implies that $a \leq Id$.
     - for any $a, b, c \in \mathcal{A}$ and any variables x, y, z, if $a(x, y), b(y, z), c(z, x) \in DC(\Theta)$ then $a; b \wedge c^{\smile} \neq 0$.
   - Let Θ be a set of quantifier-free formulas and let ϕ be any quantifier-free $L_{\infty\omega}(\mathcal{A})$-formula. We'll write $\Theta \vdash \phi$ if $\Theta \cup \{\phi\}$ is inconsistent.
   - A *quantifier free type* (or qf-type for short) $\tau(\bar{x})$ is a maximal consistent set of atomic formulas using only the variables $\bar{x}$. [We do not include negated atomic formulas in τ, but recall that the language $L(\mathcal{A})$ has one binary predicate symbol for each element of $\mathcal{A}$. So if $a(x, y)$ is an atomic L-formula then so is $(-a)(x, y)$.]

   Using these definitions we can equivalently express θ in disjunctive normal form.

   $$\theta(y, \bar{x}) \equiv \bigvee\{\bigwedge \tau(y, \bar{x}) : \text{qf-types } \tau(y, \bar{x}), \tau(y, \bar{x}) \vdash \theta(y, \bar{x})\}.$$

   To see this, it is clear that the right hand side implies the left hand side. Conversely, suppose h is an assignment to a domain A such that $A, h \models \theta(y, \bar{x})$. The set τ of all atomic formulas $\rho(\bar{z})$ with $\bar{z} \subseteq \{y\} \cup \bar{x}$ such that $A, h \models \rho(\bar{z})$ is a quantifier-free type and $\theta \in \tau$. Therefore $\tau \vdash \theta$, and hence the right hand side is also true under h.
   So

   $$\exists y\ \theta(y, \bar{x}) \equiv \bigvee\{\exists y \bigwedge \tau(y, \bar{x}) : \tau(y, \bar{x}) \vdash \theta(y, \bar{x})\}.$$

   Thus it is required to prove that $\exists y \bigwedge \tau(y, \bar{x})$ is equivalent to a quantifier-free formula for any quantifier-free type $\tau(y, \bar{x})$. If $X \not\models \exists y \exists \bar{x} \bigwedge \tau(y, \bar{x})$ then $\exists y \bigwedge \tau(y, \bar{x}) \equiv -$, over X — certainly quantifier-free.

So suppose for some $b, \bar{a} \in X$ we have

$$X \models \bigwedge \tau(b, \bar{a}) \tag{1}$$

Let $\tau \restriction_{\bar{x}} (\bar{x}) \subseteq \tau(y, \bar{x})$ be the subset of $\tau(y, \bar{x})$ consisting of those formulas that do not use the variable y. Clearly $\tau \restriction_{\bar{x}} (\bar{x})$ is a quantifier-free type.

Claim : $\exists y \bigwedge \tau(y, \bar{x})$ is equivalent, over X, to $\bigwedge \tau \restriction_{\bar{x}} (\bar{x})$.

Proof of claim: Let $\bar{c} \in X$. Trivially, if $X \models \exists y \bigwedge \tau(y, \bar{c})$ then $X \models \bigwedge \tau \restriction_{\bar{x}} (\bar{c})$. For the converse, suppose

$$X \models \bigwedge \tau \restriction_{\bar{x}} (\bar{c}).$$

Since, by 1, we also have $X \models \bigwedge \tau \restriction_{\bar{x}} (\bar{a})$ it follows that the map $\rho : \bar{a} \mapsto \bar{c}$ is a local isomorphism. By the homogeneity of X, ρ extends to a base-automorphism ρ^+ of X. Thus $X \models \bigwedge \tau(\rho^+(b), \rho^+(\bar{a}))$ and therefore $X \models \exists y \ \bigwedge \tau(y, \bar{c})$. Hence $\exists y \bigwedge \tau(y, \bar{x})$ is equivalent to the quantifier-free formula $\bigwedge \tau \restriction_{\bar{x}} (\bar{x})$, as required.

2. The proof is similar but with a finite relation algebra, for any first-order, quantifier-free formula $\theta(\bar{x})$ there are only finitely many quantifier-free types $\tau(\bar{x})$ with $\tau \vdash \theta$. Also $\tau(\bar{x})$ is a finite set so $\bigwedge \tau(\bar{x})$ is a finite conjunction. In this way we can prove that any formula is equivalent, over a homogeneous representation, to a quantifier-free formula with only finite conjunctions and disjunctions i.e. a first-order quantifier-free formula.

□

While we are thinking about quantifier elimination we include another theorem which gives a connection between relation algebras and 3 variable logic. Here we do not assume that a representation is homogeneous.

THEOREM 8 *Let $\mathcal{A}$ be a relation algebra and let $L(\mathcal{A})$ be the corresponding first-order language. Any formula $\phi(u, v, w)$ is uniformly equivalent to a quantifier-free formula $\theta(u, v, w)$. That is, the two formulas are equivalent in any representation of $\mathcal{A}$.*

PROOF:

Again, the only non-trivial case is the quantified formula $\exists v, \theta(u, v, w)$ where, inductively, $\theta(u, v, w)$ can be taken to be a quantifier-free formula. Now θ has an equivalent θ' in disjunctive normal form — a disjunction of clauses each of which is a conjunction of atomic formulas. As before, negated atoms $\neg a(u, v)$ get replaced by equivalent atomic formulas $(-a)(u, v)$.

Further, within one clause of θ', we can find an equivalent clause such that for any pair of variables (say u, v) there is only one atomic subformula involving both u and v. To see this, first replace an atomic formula $a(v, u)$ by the equivalent $a^{\smile}(u, v)$. Then replace the conjunction $a(u, v) \wedge b(u, v)$ by $(a \wedge b)(u, v)$. Thus we can find $a_i, b_i, c_i, e_i, f_i, g_i \in \mathcal{A}$ such that

$$\exists v \theta(u, v, w) \equiv \bigvee_i \exists v [a_i(u, v) \wedge b_i(v, w) \wedge c_i(u, w) \wedge e_i(u, u) \wedge f_i(v, v) \wedge g_i(w, w)]$$

Hence

$$\begin{aligned} \exists v \theta(u, v, w) &\equiv \bigvee_i [c_i(u, w) \wedge e_i(u, u) \wedge g_i(z, z) \wedge \exists v (a_i(u, v) \wedge b_i(v, w) \wedge f_i(v, v))] \\ &\equiv \bigvee_i c_i(u, w) \wedge e_i(u, u) \wedge g_i(w, w) \wedge (a_i ; (f_i \wedge Id) ; b_i)(u, w) \end{aligned}$$

which is quantifier-free. □

Similarly, for an n-dimensional cylindric algebra $\mathcal{C}$, every $L(\mathcal{C})$-formula $\phi(w_0, \ldots, w_{n-1})$ using only n variables is equivalent to a quantifier-free formula.

4.2 Complexity of NSP over $\mathcal{L}$

Now that we have a representation of the left-linear algebra $\mathcal{L}$ and now that transitive representations have been defined we return to the complexity of the NSP over $\mathcal{L}$.

THEOREM 9 *There is an algorithm with quintic complexity that decides whether an $\mathcal{L}$-network is satisfiable or not.*

PROOF:

Let N be an $\mathcal{L}$-network. We give an upper bound on the complexity of the network satisfaction problem in terms of the number of nodes in N, say $|N| = n$. If N is satisfiable then there is an embedding h mapping N into some representation X of $\mathcal{L}$. So $h(N) = \{h(m) : m \in N\} \subseteq X$ is a finite substructure of X, effectively a finite, left-linear partial order. There are two possible types of left-linear partial orders: either $h(N)$ is path connected in which case it has a minimal element x_0 say; or $h(N)$ splits into two non-empty components $h(N) = h(P) \dot{\cup} h(Q)$ and every element of $h(P)$ is incompatible ($\#$) with each element of $h(Q)$.

Suppose first that $h(N)$ is path-connected so there is some $a \in N$ with $h(a) = x_0$. If a maps to this minimal element x_0 every other element of N maps either to x_0 or to a point strictly bigger than x_0. For any node b of N, if it cannot be greater than a, then it must necessarily also map to x_0.

Definition Let $S \subseteq N$. Define

-
$$E(S) = S \cup \{m \in N : \exists s \in S, > \notin N(m, s)\}$$
$E(S)$ can be calculated from S in $O(n^2)$ time. If each element of S maps under h to the minimal element x_0 then each element of $E(S)$ must also map to x_0.
- Let $E^{i+1}(S) = E(E^i(S))$.
- Define
$$E^*(S) = \bigcup_{i<\omega} E^i(S)$$
To calculate $E^*(S)$ it suffices to let i go up to $n = |N|$ only as the sequence $S, E(S), \ldots E^i(S), \ldots$ is monotonic and must have reached a fixpoint in n steps. So $E^*(S)$ can be calculated in $O(n^3)$ time.

Now if h is an embedding of N in X mapping a to a minimal element x_0, then each point in $E^*(\{a\})$ (written $E^*(a)$ henceforth) must also map to x_0. For every other point m from N the relation from m to each $e \in E^*(a)$ must include $>$ (by the definition of $E(S)$). Therefore if x^- is any point in X with $x^- < x_0$ then the map h' which is identical to h on $N \setminus E^*(a)$ but maps $E^*(a)$ to x^-, is a consistent embedding of N in X. Thus if there is an embedding which maps a to a minimal point, then there is also an embedding which maps $E^*(a)$ to a minimal point x^- and every other node of N maps to a point strictly bigger than x^-.

Let us define a unary predicate $L(x)$ on the nodes of N thus:

$$L(a) \equiv (\forall e, e' \in E^*(a), Id \in N(e, e'))$$

We have seen that if a maps to a minimal point x_0 then $L(a)$ holds. Calculating $L(a)$ takes $O(n^3) + O(n^2) = O(n^3)$ time and calculating the validity of $\exists a \in N, L(a)$ takes $O(n^4)$ time.

Corresponding to the second alternative, where N' is not path connected, we introduce the predicate $\text{Split}(N, P, Q)$. $\text{Split}(N, P, Q)$ holds if P and Q are disjoint, non-empty subnetworks of N whose nodes jointly cover N (so N is the disjoint union of P and Q) and for all $p \in P, q \in Q$ we have $\# \in N(p, q)$.

Evaluating $(\exists P, Q \subset N)\text{Split}(N, P, Q)$ can be done in $O(n^2)$ time as follows. Form a symmetric graph $N^\#$ with the same nodes as N and with an edge (a, b) if and only if $\# \notin N(a, b)$ i.e. let there be an edge from a to b if and only if N insists that a and b are linearly ordered in some way. Now observe that N splits in some way if and only if $N^\#$ is not connected — which can be calculated in $O(n^2)$ time. If N does split, components P, Q can be found by letting N be the disjoint union of P and Q where P and Q are disconnected in $N^\#$.

This leads to the following definition, which is used to define an algorithm to test satisfiability.

$$
\begin{aligned}
|N| \leq 1 \quad &\rightarrow \quad Sat(N) && (2)\\
(\forall a \in N)\,[L(a) \quad &\rightarrow \quad (Sat(N) \leftrightarrow Sat(N \setminus E^*(a)))] && (3)\\
(\forall P, Q \subset N)\,[\text{Split}(N, P, Q) \quad &\rightarrow \quad (Sat(N) \leftrightarrow Sat(P) \wedge Sat(Q))] && (4)\\
[\neg(\exists a \in N)\; L(a) \wedge \quad & && \\
\neg(\exists P, Q \subset N)\; \text{Split}(N, P, Q)] \quad &\rightarrow \quad \neg Sat(N) && (5)
\end{aligned}
$$

An algorithm to test satisfiability is obtained by taking a network N with $|N| > 1$ and seeing if it matches the left hand side of formula 5 in which case it is not satisfiable. Otherwise it matches the antecedent of either formula 3 or 4. If it matches the left hand side of formula 3 for some $a \in N$ then, recursively, check the consistency of $N \setminus E^*(a)$. The result of this consistency check decides the consistency of N. Finally, if N satisfies the left hand side of formula 4 for some P, Q then recursively check the consistency of P and Q to decide the consistency of N. Note that there may be several different $a \in N$ matching the antecedent to formula 3 and several P, Q matching the antecedent to formula 4, but it doesn't matter which choice is made for the recursive call, as we have two-way implications in the definition of Sat so if *any* a matches the left hand side of formula 3 and the recursive call leads to the conclusion $\neg Sat(N)$ then there is no need to backtrack and try other nodes b or to try matching with formula 4.

To see that the algorithm is correct suppose, inductively, for any network N of size less than k that $Sat(N)$ holds if and only if N is satisfiable. This is true for $k = 1$. Now let N be a network with $k > 1$ nodes. If N is satisfiable let h be an embedding of N into some representation X. Either h maps N to some connected structure which means that there is some least point a and inductively $Sat(N \setminus E^*(a))$ holds. So, by formula 3, it follows that $Sat(N)$ holds too. Or h maps N to a disconnected structure, in which case N is a disjoint union of some P and Q with Split (N, P, Q) and inductively $Sat(P)$ and $Sat(Q)$ hold. By formula 4 $Sat(N)$ must hold in this case too.

Conversely, if $Sat(N)$ holds then either the antecedent of formula 3 or 4 holds. For formula 3 we have $L(a) \wedge Sat(N \setminus E^*(a))$ for some $a \in N$. Inductively, there is an assignment $h : N \setminus E^*(a) \rightarrow X$ for some representation X. We can extend the assignment h to N by mapping all the nodes in $E^*(a)$ to some point in X less than all the points $h(E^*(a))$. That such a point exists follows from the fact that X is connected and has no endpoints. Thus we have an assignment for N.

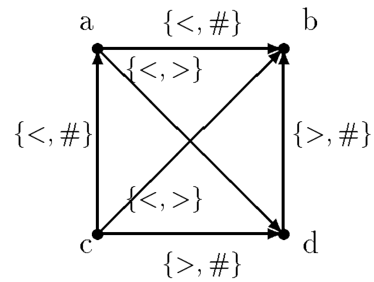
In formula 4, if $\text{Split}(N, P, Q)$ and $Sat(P) \wedge Sat(Q)$ holds for some $P, Q \subseteq N$ then, inductively again, there are assignments α, β mapping P and Q respectively into some representations. By the note on page 17 we can suppose that $\alpha : P \rightarrow X,\;\; \beta : Q \rightarrow X$ for some 2-transitive representation X. Here we only need 1-transitivity. Let p^- be any point in X less than all the points in $\alpha(P)$ and let q^- be a point less than each point in $\beta(Q)$. Find any point $p^\#$ such that q^- and $p^\#$ are incompatible $((q^-, p^\#) \in X(\#))$.

The mapping $\lambda : p^- \mapsto p^\#$ is a local isomorphism of size one. By transitivity, this extends to a base-automorphism λ^+ of X. Now, define an embedding h of N in X to be $(\alpha \circ \lambda^+) \cup \beta$. h maps N so that points from P are all incompatible with points from Q, but the relations between points in P is the same as it was under α and the relation between points in Q is given by β. Thus we have a consistent assignment to N.

Now we check the complexity of the algorithm. Formula 4 replaces the problem of finding whether N is satisfiable or not by a test on the consistency of two subnetworks. So at a given stage the algorithm may have several networks under consideration, N_i $i < k$ say. Define a non-negative variable $s = \sum_{i<k}(|N_i| - 1)$. Initially $s = |N| - 1$. Each operation of the algorithm reduces the value of s. Each step has complexity at most $O(n^4)$ and so the complexity overall is bound by $O(n^5)$.

□

In view of equation 5 in the definition of *Sat* in the proof of the previous theorem, it is quite easy to find an $\mathcal{L}$-network that is transitively closed but not satisfiable.



Indeed, for each $k > 0$ we can construct a k-consistent, unsatisfiable $\mathcal{L}$-network L. The nodes of L are $n_0, \ldots, n_{k-1}$. The labeling is defined by $L(n_i, n_{i+1}) = \{<, \#\}$ $(i < k - 1)$, $L(n_{k-1}, n_0) = \{<, \#\}$ and for $i, j < k$ with $|i - j| \neq 0, 1 (\text{mod } n)$, let $L(n_i, n_j) = \{<, >\}$. If L_i is the subnetwork of L obtained by the removal of a single node n_i then L_i has a least node, namely n_{i+1} and all subnetworks of L_i also have least nodes. Therefore L_i is satisfiable and so L is k-consistent. However L is unsatisfiable as it has no least node and cannot split into two components.

This is of some interest as the most common technique for showing that the network satisfaction problem is tractable is to show for some fixed k, that k-consistency implies satisfiability. For the relation algebra $\mathcal{L}$, however, there is no such k. Nevertheless the network satisfaction problem has quintic complexity (at worst) for $\mathcal{L}$.

5 Interpretations

The concept of an interpretation, quite well-known to model theorists, is used here to formalise how one relation algebra is capable of expressing another, and it is also an important tool for problem reduction. Thus if a representation X of one relation algebra has an interpretation in a representation Y of a second relation algebra, then we can reduce the satisfaction problem over X to that of Y. So interpretations form the principal tool for taking a known complexity result for a fixed relation algebra and generalising the result to several other relation algebras.

The reader who wishes to find out more about interpretations is referred to [Hod95]. Many of the definitions and some of the results in this section are taken from Hodges' book and made appropriate here for algebraic logic.

Interpretations have been used very successfully before with relation algebras. For example in [AGN94] representable relation algebras are interpreted in semi-groups to prove that the equational theory of various classes of relation algebras is undecidable. The equational theory under consideration there is written in the language of relation algebras i.e. the first-order language with constant and function symbols $(0, 1, \vee, -, Id, \smile, ;)$. In this article we fix one relation algebra $\mathcal{A}$ and consider properties that may hold in the representations of $\mathcal{A}$, for example which networks embed in a representation of $\mathcal{A}$. So the language under consideration here is the first-order, relational language $L = L(\mathcal{A})$. We also associate an L-theory $T_{\mathcal{A}}$ (see section 2.3). The representations of the algebra $\mathcal{A}$ coincide with the models of the theory $T_{\mathcal{A}}$. Similar languages and definitions were given in section 2.3 corresponding to a fixed, finite-dimensional cylindric algebra. Below we define an interpretation between models of such theories. Interpretations can easily be generalised to cover languages with constants and function symbols, but this is not required here.

Definitions

- Let K and L be relational languages, let Σ_K be a K-theory and Σ_L be an L-theory. Let M, N be models of Σ_K, Σ_L respectively. An *n-dimensional interpretation* Γ of M in N is a triple (τ, ∂, f) where τ is a *translation map* of the atomic K-formulas into L-formulas, $\partial \in L$ is the *domain formula* and f is a *co-ordinate map*. Formally,

  1. for every atomic formula $r(x_0, \ldots, x_{\rho-1}) \in K$, τ assigns a formula $\tau(r)(\bar{x_0}, \ldots, \bar{x_{\rho-1}}) \in L$, where $\bar{x_0}, \ldots, \bar{x_{\rho-1}}$ are each n-tuples of mutually distinct variables and ρ is the rank of r. Throughout, if x, y are distinct variables then $\bar{x}, \bar{y}$ will denote n-tuples of mutually distinct variables.
  2. the L-formula $\partial(\bar{x})$ ($\bar{x}$ is an n-tuple of variables) defines the *domain* of the interpretation. We write $\partial(N^n)$ for $\{\bar{a} \in N : N \models \partial(\bar{a})\}$.
  3. f is a surjection $\partial(N^n) \to M$

  such that for all $r \in K$, and all $\bar{a_0}, \ldots, \bar{a_{\rho-1}} \in \partial(N^n)$ we have

  $$M \models r(f(\bar{a_0}), \ldots, f(\bar{a_{\rho-1}})) \Leftrightarrow N \models \tau(r)(\bar{a_0}, \ldots, \bar{a_{\rho-1}}). \qquad (6)$$

- The pair (τ, ∂) constitute a *translation* from K to L and if $\Gamma' = (\tau, \partial, f')$ is any interpretation of a K-structure in an L-structure we'll say that Γ' *is based on* the translation (τ, ∂).

- If the translation map τ is *recursive* we'll say that any interpretation Γ based on (τ, ∂) is recursive. If K is a finite language then any translation map from K to L is recursive.

A translation map can be extended so that it is defined on arbitrary K-formulas, not just atomic ones.

LEMMA 10 *([Hod95], theorem 5.3.2.) Let $\Gamma = (\tau, \partial, f)$ be an n-dimensional interpretation of M in N. For every K-formula $\phi(x_0, x_1, \ldots, x_k)$ there is an L-formula $\tau(\phi)(\bar{x_0}, \bar{x_1}, \ldots, \bar{x_k})$ such that for all $\bar{a_0}, \ldots, \bar{a_k} \in N$*

$$N \models \tau(\phi)(\bar{a_0}, \ldots, \bar{a_k}) \Leftrightarrow M \models \phi(f(\bar{a_0}), \ldots, f(\bar{a_k})).$$

PROOF:

If ϕ is an atomic K-formula then $\tau(\phi)$ is already defined by the interpretation. For the other cases let us define

- $\tau(\neg\phi) = \neg(\tau(\phi))$
- $\tau(\phi \vee \psi) = \tau(\phi) \vee \tau(\psi)$ and
- $\tau(\exists x \phi(x, y_0, \ldots, y_k)) = \exists \bar{x}\ (\partial(\bar{x}) \wedge \tau(\phi)(\bar{x}, \bar{y_0}, \ldots, \bar{y_k}))$

Proving the lemma is just a matter of unfolding the definitions of an interpretation.
□

Now let $\mathcal{A}$ and $\mathcal{B}$ be relation algebras and let $K = L(\mathcal{A}), L = L(\mathcal{B})$. Let X, Y be representations of $\mathcal{A}, \mathcal{B}$ respectively. (So X, Y correspond to the models M, N in the definition of an interpretation.) The atomic K-formulas are all of the form $a(v, w)$ for $a \in \mathcal{A}$, some pair of variables v, w. Applying the translation gives us the $L(\mathcal{B})$-formula $\tau(a)(\bar{v}, \bar{w})$. Any interpretation has to preserve the relation algebraic operations, relative to the domain of the interpretation.

LEMMA 11 *Let $\Gamma = (\tau, \partial, f)$ be an n-dimensional interpretation of X in Y. For all $a \in \mathcal{A}, \bar{y}, \bar{z} \in \partial(Y^n)$*

$$\begin{array}{lcl} Y & \not\models & \tau(0)(\bar{y}, \bar{z}) \\ Y & \models & \tau(1)(\bar{y}, \bar{z}) \rightarrow (\tau(a)(\bar{y}, \bar{z}) \leftrightarrow \neg\tau(-a)(\bar{y}, \bar{z})) \\ Y & \models & \tau((a \vee b))(\bar{y}, \bar{z}) \leftrightarrow (\tau(a)(\bar{y}, \bar{z}) \vee \tau(b)(\bar{y}, \bar{z})) \\ Y & \models & \tau(a^{\smile})(\bar{y}, \bar{z}) \leftrightarrow \tau(a)(\bar{z}, \bar{y}) \\ Y & \models & \tau(a;b)(\bar{y}, \bar{z}) \leftrightarrow \exists\bar{x}(\partial(\bar{x}) \wedge \tau(a)(\bar{y}, \bar{x}) \wedge \tau(b)(\bar{x}, \bar{z})) \end{array}$$

PROOF:

Exercise. □

COROLLARY 12 *Let $\Gamma = (\tau, \partial, f)$ be an n-dimensional interpretation of X in Y. $\sim\, = Y(\tau(Id_{\mathcal{A}}))$ is an equivalence relation on $\partial(Y^n)$. Letting $[\bar{y}]$ denote the equivalence class of $\bar{y} \in \partial(Y^n)$ the map $a \mapsto \{([\bar{y}], [\bar{y'}]) \in \partial(Y^n)/\sim : Y \models \tau(a)(\bar{y}, \bar{y'})\}$ is a representation of $\mathcal{A}$ with domain $\partial(Y^n)/\sim$.*

A recursive interpretation 'preserves undecidability' of the satisfaction problem over a fixed representation in the following sense.

THEOREM 13 *Let $\mathcal{A}$ and $\mathcal{B}$ be relation algebras. If X, Y are representations of $\mathcal{A}, \mathcal{B}$ respectively, $\Gamma = (\tau, \partial, f)$ is an n-dimensional, recursive interpretation of X in Y and if the problem of deciding if an $L(\mathcal{A})$-formula is satisfiable in X (the satisfaction problem for $L(\mathcal{A})$ over X) is undecidable then the satisfaction problem for $L(\mathcal{B})$ over Y is undecidable too.*

PROOF:

Let $\phi(v_0, \ldots, v_{k-1}) \in L(\mathcal{A})$. Define the $L(\mathcal{B})$-formula ϕ^* to be

$$\bigwedge_{i<k} \partial(\bar{v_i}) \wedge \tau(\phi)(\bar{v_0}, \ldots, \bar{v_{k-1}}).$$

We'll show that ϕ is satisfiable in X if and only if ϕ^* is satisfiable in Y. Thus the satisfaction problem over Y must also be undecidable. Well, ϕ is satisfiable in X if and only if there exist $x_0, \ldots x_{k-1} \in X$ such that $X \models \phi(x_0, \ldots x_{k-1})$. Pick n-tuples $\bar{y_0}, \ldots, \bar{y_{k-1}} \in \partial(Y^n)$ with $f(\bar{y_i}) = x_i$ $(i < k)$ and from the definition of an interpretation

$$X \models \phi(x_0, \ldots x_{k-1}) \Leftrightarrow Y \models \phi^*(\bar{y_0}, \ldots, \bar{y_{k-1}})$$

Conversely if $Y \models \phi^*(\bar{y_0}, \ldots, \bar{y_{k-1}})$ for some $\bar{y_0}, \ldots, \bar{y_{k-1}} \in Y$ then $\bar{y_i} \in \partial(Y^n)$ which implies that ϕ is satisfiable in X at $f(\bar{y_0}), \ldots, f(\bar{y_{k-1}})$. □

Definition

- A translation $(\tau, \partial) : L(\mathcal{A}) \rightarrow L(\mathcal{B})$ is called *left-total* if for every representation Y of $\mathcal{B}$ there is a representation X of $\mathcal{A}$ and an interpretation Γ of X in Y with Γ based on (τ, ∂). If, for every representation Y of $\mathcal{B}$ there is a representation X of $\mathcal{A}$ and an n-dimensional interpretation Γ of X in Y with Γ based on (τ, ∂), then we shall call (τ, ∂) an n-dimensional, left-total translation: $L(\mathcal{A}) \rightarrow L(\mathcal{B})$.
- $(\tau, \partial) : L(\mathcal{A}) \rightarrow L(\mathcal{B})$ is called *right-total* if for every representation X of $\mathcal{A}$ there is an interpretation Γ, based on (τ, ∂), of X in some representation of $\mathcal{B}$.
- (τ, ∂) is called *total* if it is both left and right-total.
- With minor adjustments, the definitions of left-total, right-total and total apply to finite-dimensional cylindric algebras too.

THEOREM 14 *(c.f. [Hod95] theorem 5.3.3.)*
Let (τ, ∂) be an n-dimensional translation $L(\mathcal{A}) \to L(\mathcal{B})$.

1. *If (τ, ∂) is left-total then*

$$T_{\mathcal{B}} \models (\partial(\bar{u}) \wedge \partial(\bar{v})) \to [\tau(a \vee b)(\bar{u}, \bar{v}) \leftrightarrow (\tau(a) \vee \tau(b))(\bar{u}, \bar{v}) \tag{7}$$

$$\wedge \quad \tau(1_{\mathcal{A}})(\bar{u}, \bar{v}) \to (\tau(-a) \leftrightarrow \neg\tau(a))\,(\bar{u}, \bar{v}) \tag{8}$$

$$\wedge \quad (\bigwedge_{i<n} Id_{\mathcal{B}}(u_i, v_i)) \to \tau(Id_{\mathcal{A}})(\bar{u}, \bar{v}) \tag{9}$$

$$\wedge \quad \tau(a^{\smile})(\bar{u}, \bar{v}) \leftrightarrow \tau(a)(\bar{v}, \bar{u}) \tag{10}$$

$$\wedge \quad \tau(a; b)(\bar{u}, \bar{v}) \leftrightarrow \exists\bar{w}(\partial(\bar{w}) \wedge \tau(a)(\bar{u}, \bar{w}) \wedge \tau(b)(\bar{w}, \bar{v}))] \tag{11}$$

2. *Conversely, if (τ, ∂) is any n-dimensional translation satisfying 7 to 11 then the translation is left-total. Thus if Y is any representation of $\mathcal{B}$ then there is a representation Y_τ of $\mathcal{A}$ and an interpretation $\Gamma = (\tau, \partial, f)$ based on (τ, ∂) of Y_τ in Y. Furthermore, if X is any representation of $\mathcal{A}$ and $\Gamma' = (\tau, \partial, f')$ is any interpretation based on (τ, ∂) of X in Y then there is a base-isomorphism $\iota : Y_\tau \to X$ such that $\iota(f(\bar{y})) = f'(\bar{y})$ for all $\bar{y} \in \partial(Y)$.*

PROOF:

Now the first part really is a consequence of lemma 11. Here we check only conditions 9 and 11. For condition 9 let Y be any representation of $\mathcal{B}$ and let $\bar{x}, \bar{y}$ be arbitrary n-tuples from $\partial(Y^n)$. By left-totality there is a representation X of $\mathcal{A}$ and an interpretation $\Gamma = (\tau, \partial, f)$ of X in Y. If the antecedent of condition 9 is satisfied then $\bar{x} = \bar{y}$. Because $\bar{x}, \bar{y} \in \partial(Y^n)$, f is defined on $\bar{x}$ and $\bar{y}$ and $f(\bar{x}) = f(\bar{y})$. Hence $X \models Id_{\mathcal{A}}(f(\bar{x}), f(\bar{y})$ and by the definition of interpretation, it follows that $Y \models \tau(Id_{\mathcal{A}})(\bar{x}, \bar{y})$ as required.

Similarly for condition 11, let $\bar{x}, \bar{y} \in \partial(Y^n)$. Working right to left, suppose there is a tuple of points $\bar{z} \in \partial(Y^n)$ with $Y \models \tau(a)(\bar{x}, \bar{z}) \wedge \tau(b)(\bar{z}, \bar{y})$. Then $X \models a(f(\bar{x}), f(\bar{z})) \wedge b(f(\bar{z}), f(\bar{y}))$ and so $X \models (a; b)(f(\bar{x}), f(\bar{y}))$. Hence $Y \models \tau(a; b)(\bar{x}, \bar{y})$. Working in the other direction, let $\bar{x}, \bar{y} \in \partial(Y^n)$, $Y \models \tau(a; b)(\bar{x}, \bar{y})$. Then $X \models (a; b)((f(\bar{x}), f(\bar{y}))$ so there is a point $z \in X$ with $X \models a(f(\bar{x}), z) \wedge b(z, f(\bar{y}))$. Therefore (as f is surjective) there exists $\bar{z} \in \partial(Y^n), f(\bar{z}) = z$ so $Y \models \tau(a)(\bar{x}, \bar{z}) \wedge \tau(b)(\bar{z}, \bar{y})$. This gives us the implication from left to right and proves 11.

For the second part, let (τ, ∂) be a translation satisfying 7 to 11 and let Y be a representation of $\mathcal{B}$. Observe that $\sim\, = Y(\tau(Id_{\mathcal{A}}))$ is an equivalence relation on $\partial(Y^n)$ (by 9, 10 and 11). For $\bar{y} \in \partial(Y^n)$, let $[\bar{y}]$ denote the $\sim$-equivalence class of $\bar{y}$. The required representation Y_τ of $\mathcal{A}$ has domain $\partial(Y^n)/\sim$ and for all $a \in \mathcal{A}$,

$$Y_\tau(a) = \{([\bar{y_1}], [\bar{y_2}]) \in \partial(Y^n)/\sim\, : (\bar{y_1}, \bar{y_2}) \in Y(\tau(a))\}$$

To define the interpretation Γ, let $f(\bar{y}) = [\bar{y}]$. Use the formulas to show that this is well-defined and that Y_τ is a representation.

For the last part, if Γ' is an interpretation (τ, ∂, f') of X in Y then the equation $\iota(f(\bar{y})) = f'(\bar{y})$ is well-defined and suffices to define a base-isomorphism $\iota : Y_\tau \to X$. □

5.1 Expressive Power

Definition $\mathcal{B}$ *expresses* $\mathcal{A}$ *in n-dimensions* if there is an n-dimensional, total translation (τ, ∂) from $\mathcal{A}$ to $\mathcal{B}$.

Note Because we require that the translation (τ, ∂) is total, the definition of expressivity effectively quantifies over *all representations* of $\mathcal{A}$. In other research in this area, however, it is often the case that the expressive power of a relation algebra over a *restricted class* of representations or indeed a *single* representation is considered. Several authors have studied the expressive power of some of the examples below over *linear flows of time* or *intervals* from a linear flow (e.g. [VKvB89]).

If $\mathcal{X}$ and $\mathcal{Y}$ are classes of representations of $\mathcal{A}, \mathcal{B}$ respectively then we can generalise the definitions so that a translation (τ, ∂) is left total over $(\mathcal{X}, \mathcal{Y})$ if every representation in $\mathcal{Y}$ is the interpretation (based on (τ, ∂)) of a representation in $\mathcal{X}$. There are similar definitions for right total and total translations over $(\mathcal{X}, \mathcal{Y})$. $\mathcal{Y}$ expresses $\mathcal{X}$ in n dimensions if there is an n-dimensional translation (τ, ∂), total over $(\mathcal{X}, \mathcal{Y})$.

5.2 Examples of Interpretations

All the relation algebras used here are defined in section 3.2.

1. Consider the point algebra $\mathcal{P}$ and the Allen interval algebra $\mathcal{I}$. Let X be the natural representation of $\mathcal{P}$ over the rationals and let Y be the representation of $\mathcal{I}$ as ordered pairs of rational numbers. There are several possible one-dimensional interpretations of X in Y. For example define $\Gamma_1 = (\tau_1, \partial_1, f_1)$ by

$$\begin{array}{llll} \tau_1: & Id_{\mathcal{P}} & \mapsto & \{Id_{\mathcal{I}}, \text{ends}, \text{ends}^{\smile}\} & \text{(ends together)} \\ & < & \mapsto & \{\text{precedes}, \text{meets}, \text{overlaps}, \text{starts}, \text{during}\} & \text{(ends before)} \\ & > & \mapsto & \{\text{precedes}^{\smile}, \text{meets}^{\smile}, \text{overlaps}^{\smile}, \text{starts}^{\smile}, \text{during}^{\smile}\} & \text{(ends after)} \end{array}$$

   and a non-atomic element is mapped to the union of the images of the atoms inside it. The domain formula ∂_1 is defined to be truth and the map f_1 takes an interval (p, q) and maps it to its endpoint q.

2. $\mathcal{I}$ expresses $\mathcal{P}$ in one dimension. To see this we'll show that the translation (τ_1, ∂_1) in the previous example is total. For right-totality, let X be any representation of $\mathcal{P}$, so X is a dense linear order without endpoints. Make a representation Y of $\mathcal{I}$ whose domain consists of all pairs (x, y) from X with $(x, y) \in X(<)$ (briefly $x < y$). Then let $Y(\text{overlaps}) = \{(u, v), (x, y) \in Y : u < x < v < y\}$ and represent the other twelve atoms in a similar, natural way. As in the previous example there is a one-dimensional interpretation of X in Y, based on the same translation as in example 1. Thus (τ_1, ∂_1) is right-total.

   For left-totality check that (τ_1, ∂_1) satisfies equations 7 to 11 of theorem 14.

3. Let X, Y be the representations of $\mathcal{P}, \mathcal{I}$, respectively, as in example 1. There is a two-dimensional interpretation $\Gamma_2 = (\tau_2, \partial_2, f_2)$ of Y in X. The domain is defined by $\partial_2(v, w) = (v < w)$ and the thirteen atoms of $\mathcal{I}$ are mapped in an obvious way, for example τ_2 : $\text{meets}(u, v) \mapsto \tau_2(\text{meets})(u_0, u_1, v_0, v_1)$ where

$$\tau_2(\text{meets})(u_0, u_1, v_0, v_1) = (u_0 < u_1 = v_0 < v_1)$$

   f_2 takes a pair $x_1, x_2 \in X$ with $x_1 < x_2$ and maps them to the interval $(x_1, x_2) \in Y$.

4. Similarly, it can be shown that $\mathcal{P}$ expresses $\mathcal{I}$ in two dimensions.

5. Let X be the representation of $\mathcal{P}$ based on the rationals and let Y be the representation of the metric point algebra $\mathcal{M}$ also having domain $\mathbb{Q}$ (see page 11). There is a one-dimensional interpretation $\Gamma_3 = (\tau_3, \partial_3, f_3)$ of X in Y. To define the domain let $\partial_3(w) = (w = w)$. Translate the three atoms of $\mathcal{P}$ by

$$\begin{array}{llll} \tau_3: & Id_{\mathcal{P}} & \mapsto & [0, 0] \\ & < & \mapsto & (0, \infty) \\ & > & \mapsto & (-\infty, 0) \end{array}$$

   and let $f_3(q) = q$, for all rationals q.

6. Surprisingly perhaps, the metric point algebra $\mathcal{M}$ cannot express $\mathcal{P}$ in one dimension, at least the translation defined in part 5 won't work and it is hard to imagine that any other one will do better. Although any countable representation of $\mathcal{P}$ (and many of its other representations too) has a one dimensional interpretation in some representation of $\mathcal{M}$ this is not always the case for uncountable representations. Representations of $\mathcal{P}$ are all dense linear orders without endpoints, whereas representations of $\mathcal{M}$ have the further property that the topology generated by the open intervals has a metric. It is not, in general, the case that this topology on an arbitrary dense linear order has a metric.

7. There is a two dimensional, total translation from $\mathcal{B}_7$ to the two-atom algebra $\mathcal{A}_2$. The domain formula is $\partial(x_0, x_1) = \sharp(x_0, x_1)$ which demands that a pair of points is distinct. The translation of the seven atoms is natural, for example $\tau(\text{Swap})(x_0, x_1, y_0, y_1) = Id_{\mathcal{A}_2}(x_1, y_0) \wedge Id_{\mathcal{A}_2}(x_0, y_1)$. The other six atoms translate in a similar way.

8. The Allen interval algebra $\mathcal{I}$ cannot express the containment algebra $\mathcal{C}$ in one dimension. To indicate why this is the reader will have to glance ahead at some of the results that follow. Suppose that (τ, ∂) is a one dimensional translation of $\mathcal{C}$ in $\mathcal{I}$. It is a theorem of [LM94] that $\mathcal{I}$ has only one countable representation, up to base-isomorphism, and we will see (corollary 18) that this is homogeneous. Looking ahead to lemma 16 we can take (τ, ∂) to be a quantifier-free, one dimensional translation. Since the identity of $\mathcal{I}$ is an atom, the domain element ∂ must be equivalent to truth. Thus by lemma 15, τ defines an injection from $\mathcal{C}$ into $\mathcal{I}$ respecting the operations of converse and composition.

   Next, we claim that the only injection from $\mathcal{C}$ into $\mathcal{I}$ respecting $\smile$ and ; is the natural inclusion map (e.g. 'contained-in' maps to { starts, during, ends }.) To prove this claim observe that because Γ respects composition and converse, $\tau(1_{\mathcal{C}})$ must be an *equivalence element* e of $\mathcal{A}$ i.e. it satisfies $e = e; e = e^{\smile}$. But $\mathcal{I}$ has only the following equivalence elements: $Id_{\mathcal{I}}, 1_{\mathcal{I}}$, { starts, Id, starts$^{\smile}$} and { ends, Id, ends$^{\smile}$}. Thus $\tau(1_{\mathcal{C}})$ can only be $1_{\mathcal{I}}$ as all the other equivalence elements contain less than five atoms, τ preserves $\vee$ and is injective. Hence τ maps $\mathcal{C}$ to a *subalgebra* of $\mathcal{I}$ and it is fairly routine to check that $\mathcal{I}$ has only one subalgebra with five atoms. So for any $L(\mathcal{C})$-formula ϕ we have $\tau(\phi) = \phi$.

   Let X be a countable representation of $\mathcal{C}$ based on an atomless boolean algebra, defined on page 10. If there were a one-dimensional interpretation Γ of X in a countable representation Y of $\mathcal{I}$ then for any $L(\mathcal{C})$-formula ϕ, by lemma 10, $X \models \phi$ if and only if $Y \models \tau(\phi)$. By the forgoing $\tau(\phi) = \phi$. However, the network N given in the diagram on page 11 has the property that N embeds in X but not in Y. Thus $X \models \exists \bar{x} \phi_N(\bar{x})$ but $Y \not\models \exists \bar{x} \phi_N(\bar{x})$ for any representation Y of $\mathcal{I}$. This shows that there is no one dimensional interpretation of X in a representation of $\mathcal{I}$.

5.3 Interpreting in a homogeneous representation

If Γ interprets X in Y and Y is homogeneous then, by theorem 7, we can eliminate all quantifiers over Y. This allows an interpretation to take the special form given in the following lemma.

LEMMA 15 *Let $\mathcal{A}$, $\mathcal{B}$ be relation algebras and let $\Gamma = (\tau, \partial, f)$ be a one-dimensional interpretation of X in Y where X is any representation of $\mathcal{A}$ and Y is a homogeneous representation of $\mathcal{B}$.*

1. *For $a \in \mathcal{A}$, the $L(\mathcal{B})$-formula $\tau(a)(v, w)$ is equivalent, over Y, to an atomic $L(\mathcal{B})$-formula $b(v, w)$ for some $b \in \mathcal{B}$. Thus we can assume that for each $a \in \mathcal{A}$, $\tau(a)(v, w)$ is an atomic $L(\mathcal{B})$-formula and $\tau(a) \in \mathcal{B}$.*

2. *The formula $\partial(w)$ is equivalent over Y to $\delta(w, w)$ for some $\delta \leq Id_{\mathcal{B}} \in \mathcal{B}$.*

3. *If $\delta = Id_{\mathcal{B}}$ in 2 (above) then the map $g : a \mapsto \tau(a)$ is an injection from $\mathcal{A}$ into $\mathcal{B}$ preserving the operations $0, \vee, ^{\smile}$ and ;.*

PROOF:

The first two parts follow from theorem 7. Part 3 follows from lemma 11 and the fact that $\partial(w)$ is equivalent to truth.
□

LEMMA 16 *Let $\mathcal{B}$ be a relation algebra with a countable, homogeneous representation Y and suppose all its countable representations are base-isomorphic to Y. If (τ, ∂) is any one-dimensional, total translation from a relation algebra $\mathcal{A}$ to $\mathcal{B}$ then for all $a \in \mathcal{A}$ there is an element $b \in \mathcal{B}$ such that*

$$T_{\mathcal{B}} \models \tau(a)(v, w) \leftrightarrow b(v, w)$$

and there is a $\delta \leq Id_{\mathcal{B}} \in \mathcal{B}$ such that

$$T_{\mathcal{B}} \models \partial(w) \leftrightarrow \delta(w, w)$$

(i.e. we can uniformly eliminate quantifiers from the translation.)

PROOF:

Let Y' be any representation of $\mathcal{B}$. By the Löwenheim-Skolem theorem, Y' is elementary equivalent to the countable, homogeneous representation Y. By the left-totality of (τ, ∂) there is an interpretation (τ, ∂, f) of X in Y, for some representation X of $\mathcal{A}$. By lemma 15, for any $a \in \mathcal{A}$ there is a $b \in \mathcal{B}$ such that $Y \models \tau(a)(v, w) \leftrightarrow b(v, w)$. Since Y' is elementary equivalent to Y, it follows that $Y' \models (\forall v, w)\ \tau(a)(v, w) \leftrightarrow b(v, w)$ also. Thus the quantifier-free translation that takes a representation of $\mathcal{A}$ to Y is uniformly equivalent to (τ, ∂).
□

Let us conclude this section with a previously advertised lemma that can be used to show that the representation of the Allen interval algebra based on ordered pairs of rationals is homogeneous. First a definition.

Definition Let ∂ be some domain formula and let Y be a representation of $\mathcal{B}$. Let Θ be a set of $L(\mathcal{B})$-formulas. $\langle\Theta\rangle_{\partial(Y)}$ — the set of formulas *generated* by Θ over $\partial(Y)$ — is defined to be the smallest set of $L(\mathcal{B})$-formulas containing Θ, closed under equivalence modulo $\partial(Y)$ and closed under first-order connectives in the following sense:

$$\begin{array}{rcl} \theta \in \langle\Theta\rangle_{\partial(Y)} & \Rightarrow & \neg\theta \in \langle\Theta\rangle_{\partial(Y)} \\ \theta_1, \theta_2 \in \langle\Theta\rangle_{\partial(Y)} & \Rightarrow & \theta_1 \vee \theta_2 \in \langle\Theta\rangle_{\partial(Y)} \\ \theta(\bar{x}, \bar{y}) \in \langle\Theta\rangle_{\partial(Y)} & \Rightarrow & \exists\bar{x}(\partial(\bar{x}) \wedge \theta(\bar{x}, \bar{y})) \in \langle\Theta\rangle_{\partial(Y)} \end{array}$$

An $L(\mathcal{B})$-formula ϕ is said to be generated by Θ over $\partial(Y)$ if $\phi \in \langle\Theta\rangle_{\partial(Y)}$.

LEMMA 17 *Let Y be a homogeneous representation of $\mathcal{B}$ and let X be any representation of $\mathcal{A}$. If there is an n-dimensional interpretation $\Gamma = (\tau, \partial, f)$ of X in Y such that for all $i, j < n$*

1. *there is a (unique) atom $b_{ij} \in \mathcal{B}$ with $Y \models \partial(\bar{v}) \rightarrow b_{ij}(v_i, v_j)$ and*
2. *for each atom $b \in \mathcal{B}$ the $L(\mathcal{B})$-formula $b(v_i, w_j)$ is generated by $\{\tau(a)(\bar{v}, \bar{w}) : a \in \mathcal{A}\}$ over $\partial(Y)$*

then X is homogeneous too.

PROOF:

Let $\alpha : (x_0, \ldots, x_{k-1}) \mapsto (x'_0, \ldots, x'_{k-1})$ be any local isomorphism of X. We need to show that α can be extended to a base-automorphism α^+ of X. Now since f is a surjection, there are n-tuples $\bar{y_i}, \bar{y'_i} \in \partial(Y^n)$ $(i < k)$ with $f(\bar{y_i}) = x_i$ and $f(\bar{y'_i}) = x'_i$. Further, the $(n \times k)$-tuples $(\bar{y_0}, \ldots, \bar{y_{k-1}})$ and $(\bar{y'_0}, \ldots, \bar{y'_{k-1}})$ are locally isomorphic in Y. To see this, let γ_1, γ_2 be any two elements from the sequence $(\bar{y_0}, \ldots, \bar{y_{k-1}})$ and let γ'_1, γ'_2 be the corresponding elements from $(\bar{y'_0}, \ldots, \bar{y'_{k-1}})$. If $\gamma_1, \gamma_2 \in \bar{y_i}$ (some $i < k$) then use part 1 to show that (γ_1, γ_2) and (γ'_1, γ'_2) satisfy the same atoms and hence the same quantifier-free $L(\mathcal{B})$-formulas. Otherwise, if $\gamma_1 \in \bar{y_i}$ and $\gamma_2 \in \bar{y_j}$ with $i \neq j < k$ then use part 2 to show that (γ_1, γ_2) and (γ'_1, γ'_2) satisfy the same relations in $L(\mathcal{B})$.

By the homogeneity of Y the map $(\bar{y_0}, \ldots, \bar{y_{k-1}}) \mapsto (\bar{y'_0}, \ldots, \bar{y'_{k-1}})$ extends to a base-automorphism ϕ of Y. The required base-automorphism of X is defined by

$$\alpha^+ = f^{-1} \circ \phi \circ f.$$

To check that α^+ is well-defined, let $\bar{y}, \bar{y'} \in Y$ be such that $f(\bar{y}) = f(\bar{y'}) \in X$. Then $X \models Id_{\mathcal{A}}(f(\bar{y}), f(\bar{y'}))$ and hence $Y \models \tau(Id_{\mathcal{A}})(\bar{y}, \bar{y'})$. Since ϕ is a base-automorphism of Y, $Y \models \tau(Id_{\mathcal{A}})(\phi(\bar{y}), \phi(\bar{y'}))$. Therefore $X \models Id_{\mathcal{A}}(f(\phi(\bar{y})), f(\phi(\bar{y'})))$ and so $f(\phi(\bar{y})) = f(\phi(\bar{y'}))$. This shows that α^+ is well-defined and to show that it is a base-automorphism extending α is entirely routine. □

COROLLARY 18 *The representation Y of the Allen interval algebra, based on ordered pairs of rationals, is a homogeneous representation.*

PROOF:

Check that the two-dimensional interpretation Γ_2 (see section 5.2 example 3) of Y in the representation X of $\mathcal{P}$ based on the rationals meets the conditions in the previous lemma. For example, to express the atomic $L(\mathcal{P})$-formula $< (v_1, w_1)$, consider the $L(\mathcal{I})$-formula $\sigma(v, w) = \{\text{precedes}, \text{meets}, \text{overlaps}, \text{starts}, \text{during}\}(v, w)$. From the translation, $\tau_2(\sigma)(v_0, v_1, w_0, w_1) \equiv v_1 < w_1$. □

6 Interpretations and problem reduction

In this section we will show how interpretations can be used to take known complexity results and apply them to other algebras.

First, we tie-in the definitions of expressive power with the complexity of the satisfaction problem in the following lemma.

LEMMA 19 *Let $\mathcal{A}$ and $\mathcal{B}$ be finite relation algebras and let $\mathcal{B}$ express $\mathcal{A}$ in n dimensions. The satisfaction problem for $\mathcal{A}$* reduces to *the satisfaction problem for $\mathcal{B}$ in linear time. Thus the complexity of the satisfaction problem over $\mathcal{B}$ is at least as high as the complexity of the satisfaction problem over $\mathcal{A}$.*

PROOF:

Let (τ, ∂) be a total, n-dimensional translation from $\mathcal{A}$ to $\mathcal{B}$. For any $L(\mathcal{A})$-formula ϕ, if X and Y are representations of $\mathcal{A}, \mathcal{B}$ respectively and (τ, ∂, f) is an interpretation of X in Y then, by lemma 10, $X \models \phi$ if and only if $Y \models \tau(\phi)$. Now because (τ, ∂) is total, for every representation X of $\mathcal{A}$ there is a representation Y of $\mathcal{B}$ with an interpretation based on (τ, ∂) of X in Y and, conversely, for every representation Y of $\mathcal{B}$ there is a representation X of $\mathcal{A}$ and an interpretation of X in Y. Thus ϕ is satisfiable in a representation of $\mathcal{A}$ if and only if $\tau(\phi)$ is satisfiable in a representation of $\mathcal{B}$. $\tau(\phi)$ can be calculated from ϕ in linear time (see lemma 10). □

For infinite, but recursively definable relation algebras a similar result can be obtained but the translation τ must be recursive and consideration needs to be given to the time complexity of the translation.

6.1 Reducing the NSP for relation algebras

Now we obtain some basic results connecting a certain type of expressive power to the complexity of the network satisfaction problem. Whereas previously we translated atomic formulas $a(v,w)$ into arbitrary formulas $\tau(a)(\bar{v},\bar{w})$, in this section we want to be able to define $\tau(a)$ using networks.

Notation Let N be an $\mathcal{A}$-network with nodes $\bar{x}$. Recall that we have previously defined the quantifier-free formula $\phi_N(x_p : p \in N)$ to be $\bigwedge_{p,q\in N} N(p,q)(x_p,x_q)$, where x_p is a variable corresponding to the node p. If $\bar{q} = (q_0,\ldots,q_j)$ is a subset of the nodes of the network N then write

$$(\exists \bar{x_{\bar{q}}})\phi_N(x_p : p \in N) \stackrel{\text{def}}{=} (\exists x_{q_0},\ldots,x_{q_j})\phi_N(x_p : p \in N)$$

The quantifier $\forall \bar{x_{\bar{p}}}$ has a similar definition. Observe, for any representation X of $\mathcal{A}$ and any $\bar{q} \subseteq N$, that N is satisfiable in X if and only the formula $(\exists \bar{x_{\bar{q}}})\phi_N(x_p : p \in N)$ is satisfiable in the $L(\mathcal{A})$-structure X.

Definitions

- An $L(\mathcal{A})$-formula $\phi(\bar{x})$ is said to be *definable by an $\mathcal{A}$-network* if and only if there is an $\mathcal{A}$-network N with nodes $\bar{p} \dot\cup \bar{q}$ with $|\bar{p}| = |\bar{x}|$ and such that

$$T_{\mathcal{A}} \models (\phi(x_p : p \in \bar{p}) \leftrightarrow (\exists \bar{x_{\bar{q}}})\phi_N(x_r : r \in N))$$

- Let M,N be $\mathcal{A}$-networks possibly sharing some nodes. Define the $\mathcal{A}$-network $M \wedge N$ with nodes $M \cup N$ by

$$(M \wedge N)(p,q) \stackrel{\text{def}}{=} M(p,q) \wedge N(p,q)$$

  where $M(p,q)$ is taken to be 1 if either p or q is not a node of M and similarly for $N(p,q)$.

- Similarly, if $\{N_\lambda : \lambda \in \Lambda\}$ is a finite set of networks (not necessarily disjoint) then we can define $\bigwedge_{\lambda\in\Lambda} N_\lambda$ to be the network with nodes $\bigcup_\lambda N_\lambda$ and for any pair of nodes p,q we have $(\bigwedge_\lambda N_\lambda)(p,q) = \bigwedge_\lambda(N_\lambda(p,q))$.

LEMMA 20 *Let (τ,∂) be an n-dimensional translation from $\mathcal{A}$ to $\mathcal{B}$. Suppose that $a,b \in \mathcal{A}$ and $\tau(a),\tau(b)$ are both definable by $\mathcal{B}$-networks. Then the following $L(\mathcal{B})$-formulas are also definable by $\mathcal{B}$-networks: $\tau(a \wedge b), \tau(a^\smile), \tau(a;b)$.*

PROOF:

Let $M = M(\bar{p},\bar{q},\bar{r}), N = N(\bar{p'},\bar{q'},\bar{r'})$ be $\mathcal{B}$-networks with nodes $\bar{p} \dot\cup \bar{q} \dot\cup \bar{r}$ and $\bar{p'} \dot\cup \bar{q'} \dot\cup \bar{r'}$ respectively such that $\tau(a)(\bar{x_{\bar{p}}},\bar{x_{\bar{q}}})$ is equivalent in representations of $\mathcal{B}$ to $(\exists \bar{x_{\bar{r}}})\phi_M(\bar{x_{\bar{p}}},\bar{x_{\bar{q}}},\bar{x_{\bar{r}}})$ and $\tau(b)(\bar{x_{\bar{p'}}},\bar{x_{\bar{q'}}})$ is equivalent to $(\exists \bar{x_{\bar{r'}}})\phi_N(\bar{x_{\bar{p'}}},\bar{x_{\bar{q'}}},\bar{x_{\bar{r'}}})$.

Consider $\tau(a \wedge b)$ first. The required network to define it is $L = M(\bar{p},\bar{q},\bar{r}) \wedge N(\bar{p},\bar{q},\bar{r'})$ (where $\bar{p},\bar{q},\bar{r}$ and $\bar{r'}$ are mutually disjoint sequences of nodes). $\tau(a\wedge b)(\bar{x_{\bar{p}}},\bar{x_{\bar{q}}})$ is equivalent (by lemma 10) to $\tau(a)(\bar{x_{\bar{p}}},\bar{x_{\bar{q}}}) \wedge \tau(b)(\bar{x_{\bar{p}}},\bar{x_{\bar{q}}})$. This is assumed equivalent to $(\exists \bar{x_{\bar{r}}})\phi_M(\bar{x_{\bar{p}}},\bar{x_{\bar{q}}},\bar{x_{\bar{r}}}) \wedge (\exists \bar{x_{\bar{r'}}})\phi_N(\bar{x_{\bar{p}}},\bar{x_{\bar{q}}},\bar{x_{\bar{r'}}})$ which is equivalent, by definition of L, to $(\exists \bar{x_{\bar{r}}},\bar{x_{\bar{r'}}})\phi_L(\bar{x_{\bar{p}}},\bar{x_{\bar{q}}},\bar{x_{\bar{r}}},\bar{x_{\bar{r'}}})$. So $\tau(a \wedge b)$ is definable by a network.

$M(\bar{q},\bar{p},\bar{r})$ can be used to define $\tau(a^\smile)(\bar{x_{\bar{p}}},\bar{x_{\bar{q}}})$.

To define $\tau(a;b)$ by a network, let P be the network $M(\bar{p},\bar{q},\bar{r}) \wedge N(\bar{q},\bar{s},\bar{t})$ where all sequences of nodes are mutually disjoint. Then $\tau(a;b)(\bar{x_{\bar{p}}},\bar{x_{\bar{s}}})$ is equivalent to $(\exists \bar{x_{\bar{q}}},\bar{x_{\bar{r}}},\bar{x_{\bar{t}}})\phi_P(\bar{x_{\bar{p}}},\bar{x_{\bar{q}}},\bar{x_{\bar{r}}},\bar{x_{\bar{s}}},\bar{x_{\bar{t}}})$.

□

THEOREM 21 *1. Let $\mathcal{A}$ be a finite relation algebra and let $(\tau,\partial) : \mathcal{A} \to \mathcal{B}$ be an n-dimensional, total translation. If the domain formula $\partial(\bar{x})$ and the translated formulas $\tau(a) : a \in \mathcal{A}$ are definable by networks then the network satisfaction problem for $\mathcal{A}$ reduces in quadratic time to the network satisfaction problem for $\mathcal{B}$.*

2. *Let X be a representation of a finite relation algebra $\mathcal{A}$ and suppose that for any $\mathcal{A}$-network N, N is satisfiable in some representation of $\mathcal{A}$ if and only if N is satisfiable in X (such an X is sometimes called a universal representation). If $(\tau, \partial) : \mathcal{A} \to \mathcal{B}$ is any n-dimensional, left-total translation such that there is a representation Y of $\mathcal{B}$ and an interpretation of X in Y based on (τ, ∂) and with $\partial(\bar{x})$ and $\tau(a) : a \in \mathcal{A}$ definable by networks, then the network satisfaction problem for $\mathcal{A}$ reduces to the network satisfaction problem for $\mathcal{B}$.*

PROOF:

1. Let N be an $\mathcal{A}$-network. Let $\bar{p}$ be an n-tuple of distinct nodes, for $p \in N$. For each such p let N^p be a network with nodes $\bar{p} \cup \bar{p'}$ where $\bar{p'}$ is a (possibly empty) sequence of additional nodes, unique to p (i.e. $p \neq q \Rightarrow (\bar{p} \cup \bar{p'}) \cap (\bar{q} \cup \bar{q'}) = \emptyset$) and $\exists \bar{x_{p'}} \phi_{N^p}(\bar{x_p}, \bar{x_{p'}})$ is equivalent, in representations of $\mathcal{B}$, to $\partial(\bar{x_p})$.
   Similarly, for $p, q \in N$ let N^{pq} be a $\mathcal{B}$-network whose nodes include $\bar{p}, \bar{q}$, possibly some other nodes $\bar{r}$ and $(\exists \bar{x_r})\phi_{N^{pq}}(\bar{x_p}, \bar{x_q}, \bar{x_r})$ is equivalent, in representations of $\mathcal{B}$, to $\tau(N(p,q))(\bar{x_p}, \bar{x_q})$. The additional nodes $\bar{r} = \bar{r}(p,q)$ depend on the pair p, q and are mutually distinct. In other words, if $p' \neq p$ or $q' \neq q \in N$ then $\bar{r}(p,q) \cap \bar{r}(p',q') == \emptyset$ (all the extra nodes are distinct). Now let

   $$N^* = \bigwedge_{p \in N} N^p \wedge \bigwedge_{p,q \in N} N^{pq}.$$

   If d is a bound on $|N^p|, |N^{pq}| : p, q \in N$ then $|N^*|$ is bound by $d \times |N| + d \times |N|^2$.

   We must prove that N is satisfiable in a representation of $\mathcal{A}$ if and only if N^* is satisfiable in a representation of $\mathcal{B}$. If h is a homomorphism from N into a representation X of $\mathcal{A}$ then, because (τ, ∂) is right-total, there is a representation Y of $\mathcal{B}$ and an interpretation $(\tau, \partial, f) : X \to Y$. For each node $p \in N$ use the surjectivity of f to pick any n-tuple of points $\bar{y}(p) \in \partial(Y^n)$ such that $f(\bar{y}(p)) = h(p)$. By the definition of an interpretation $X \models N(p,q)(h(p), h(q))$ if and only if $Y \models \tau(N(p,q))(\bar{y}(p), \bar{y}(q))$ which is equivalent (by definition of N^{pq}) to $Y \models (\exists \bar{x_r})\phi_{N^{pq}}(\bar{y}(p), \bar{y}(q), \bar{x_r})$. Thus N^* is satisfiable in Y.
   Conversely, if j is a homomorphism from N^* into any representation Y of $\mathcal{B}$, use left-totality to find a representation X of $\mathcal{A}$ which interprets in Y. Define a homomorphism $h : N \to X$ by $p \mapsto f(j(\bar{p}))$, for $p \in N$. Note, by the construction of N^*, in particular the definition of N^p, that $j(\bar{p}) \in \partial(Y^n)$ so f is defined on its argument.
   Thus the network satisfaction problem for $\mathcal{A}$ reduces to the network satisfaction problem for $\mathcal{B}$.
2. Let N be any $\mathcal{A}$-network. N is satisfiable if and only if it is satisfiable in the representation X. But X interprets in a representation Y of $\mathcal{B}$. Using this fact and the left-totality of (τ, ∂) the argument used in the first part of this proof goes through.

□

Notes:

1. The complexity of the reduction in the theorem is quadratic in terms of the number of nodes of N. But in terms of the number of edges of N this becomes linear complexity. We do not dwell on this point as in this paper we are only investigating polynomial time reductions and make no finer distinction.
2. The formula $\tau(a)$ needs to be equivalent to a network $N(a)$ for *all* $a \in \mathcal{A}$, not just the atoms. For example, there is a two-dimensional interpretation of the Allen interval algebra

$\mathcal{I}$ in the linear point algebra $\mathcal{P}$ in which each of the thirteen Allen atoms gets interpreted as a network of size four, but not all non-atomic elements translate in this way. This shows that the NSP for *atomic networks* over $\mathcal{I}$ has polynomial complexity but proves nothing about the general NSP for $\mathcal{I}$. This is hardly surprising as the NSP for $\mathcal{I}$ is NP-hard while the NSP for the point algebra has cubic complexity, so there can be no reduction, unless P = NP.

3. Taking the argument a little further, using the two-dimensional interpretation of $\mathcal{I}$ in $\mathcal{P}$, the Allen relations that can be defined by networks are often called the *pointizable* relations. Consider the NSP over $\mathcal{I}$ where we impose the restriction on the networks that all the labels of the edges are pointizable interval relations. The proof of theorem 21 shows that this problem reduces to the NSP over $\mathcal{P}$ and hence has polynomial complexity. (In fact it has cubic complexity). However, research of [NB94, Lig94] shows that a very much larger set of Allen relations can be used to label networks while preserving tractability. In a separate paper we aim to investigate Ligozat's concept of *preconvexity* to generalise this work to cover relation algebras not based on a linear order.

COROLLARY 22 *If $\mathcal{B}$ expresses $\mathcal{A}$ in one dimension using a quantifier-free translation then the network satisfaction problem for $\mathcal{A}$ reduces in linear time to the network satisfaction problem for $\mathcal{B}$.*

PROOF:

We are assuming that there is a one-dimensional, quantifier-free, total translation (τ, ∂) from $\mathcal{A}$ to $\mathcal{B}$. Thus, for all $a \in \mathcal{A}$ the quantifier-free formula $\tau(a)(x, y)$ is equivalent to $b(x, y)$ for some $b \in \mathcal{B}$. This is equivalent to a network N with two nodes p, q and $N(p, q) = b$. $\Box$

6.2 Intractability of NSP for CA_n $(n > 3)$

Let $n > 3$ and let $\mathcal{C}$ be an n-dimensional cylindric algebra. In this section we'll show that, provided every satisfiable $\mathcal{C}$-network is satisfied in a representation of size at least five, the network satisfaction problem is always NP-hard over $\mathcal{C}$. Recall from theorem 6 that the NSP for $\mathcal{D}_4$ is NP-complete and from lemma 5 that any satisfiable $\mathcal{D}_4$-network is satisfied in the countable representation X_ω.

First, the reduction theorem.

THEOREM 23 *Let $\mathcal{C}$ and $\mathcal{D}$ be finite cylindric algebras. If there is a total, one-dimensional, quantifier-free translation from $\mathcal{C}$ into $\mathcal{D}$ then the network satisfaction problem for $\mathcal{C}$ reduces, in linear time, to that of $\mathcal{D}$.*

PROOF:

As corollary 22. $\Box$

So if there is a total, one-dimensional, quantifier-free translation from $\mathcal{D}_4$ into $\mathcal{C}$ for some cylindric algebra $\mathcal{C}$ then the network satisfaction problem for $\mathcal{C}$ is also NP-hard. We will prove this complexity result for virtually every cylindric algebra of dimension 4 or more, but we have to use a condition on the translation that is slightly weaker than totality. The next definition is constructed specially to help with the following theorem and is not used elsewhere.

Definition Let $n \geq 4$, let $\mathcal{A}$ and $\mathcal{B}$ be any countable, simple n-dimensional cylindric algebras and let (τ, ∂) be a translation from $L(\mathcal{A})$ to $L(\mathcal{B})$.

- (τ, ∂) is *near-left-total* if for any cubic representation Y of $\mathcal{B}$ with $|Y| > 5$ there is a representation X of $\mathcal{A}$ and an interpretation Γ of X in Y with Γ based on (τ, ∂).

- (τ, ∂) is *near-right-total* if for any infinite, cubic representation X of $\mathcal{A}$ there is a cubic representation Y of $\mathcal{B}$ and an interpretation $\Gamma : X \to Y$ based on (τ, ∂).

- (τ, ∂) is *near-total* if it is both near-left-total and near-right-total.

LEMMA 24 *Let $n \geq 4$ and let $\mathcal{C} \in \mathbf{CA}_n$ be any countable (possibly finite) simple n-dimensional cylindric algebra with an infinite, cubic representation. Then there is a one-dimensional, quantifier-free,* near-total *translation $(\tau, \partial) : \mathcal{D}_4 \to \mathcal{C}$.*

PROOF:

Since $\mathcal{C}$ has an infinite, cubic representation, by the Löwenheim-Skolem theorem, it has cubic representations of all infinite cardinalities. Let us write d_{ij} $(i, j < 4)$ for the diagonals of $\mathcal{D}_4$, and $d^{\mathcal{C}}_{ij}$ $(i, j < n)$ for the diagonals of $\mathcal{C}$. To define the translation (τ, ∂), let the domain formula $\partial(w) = (w = w)$ and

$$\tau(d_{ij}) = d^{\mathcal{C}}_{ij} \qquad (i, j < 4)$$

(since every element of $\mathcal{D}_4$ is a boolean combination of diagonals this is enough to translate every element of $\mathcal{D}_4$). For near-left-totality, let Y be any cubic representation of $\mathcal{C}$ with $|Y| > 5$. Then there is a representation X of $\mathcal{D}_4$ with $|X| = |Y|$. Define an interpretation Γ based on τ of X in Y by letting $f : Y \to X$ be any bijection. Then, for any 4-tuple $(y_0, y_1, y_2, y_3) \in Y$, any $i \neq j < 4$

$$\begin{aligned} Y \models d^{\mathcal{C}}_{ij}(y_0, \ldots, y_3) &\Leftrightarrow y_i = y_j \\ &\Leftrightarrow f(y_i) = f(y_j) \\ &\Leftrightarrow X \models d_{ij}(f(y_0), \ldots f(y_3)) \end{aligned}$$

Clearly boolean combinations of the diagonals will also be interpreted correctly, so Γ is an interpretation based on (τ, ∂).

Similarly, for any infinite, cubic representation X of $\mathcal{D}_4$ there is a representation Y of $\mathcal{C}$ with $|Y| = |X|$. As above we can find an interpretation Γ based on (τ, ∂) of X in Y.

□

THEOREM 25 *The network satisfaction problem is* **NP***-hard for any finite cylindric algebra $\mathcal{C}$ of dimension greater than 3 provided every satisfiable $\mathcal{C}$-network can be satisfied in some representation of $\mathcal{C}$ of size greater than five.*

PROOF:

The proof works be reducing the network satisfaction problem for $\mathcal{D}_4$ to the NSP over $\mathcal{C}$. By lemma 5 the NSP for $\mathcal{D}_4$ is equivalent to the NSP for $\mathcal{D}_4$ over X_ω. By lemma 24 there is a one-dimensional, quantifier-free translation $(\tau, \partial) : \mathcal{D}_4 \to \mathcal{C}$ and an interpretation $\Gamma : X_\omega \to Y$ with Γ based on (τ, ∂), for some representation Y of $\mathcal{C}$. So if N is satisfied in a representation of $\mathcal{D}_4$ then it is satisfied in X_ω and hence $\tau(N)$ is satisfied in a representation Y of $\mathcal{C}$. Note that $\tau(N)$ is, strictly, a 4-dimensional network where every 4-tuple is labelled by an element of $\mathcal{C}$. Now the dimension n of $\mathcal{C}$ may be greater than 4 and the network satisfaction problem for $\mathcal{C}$ deals with n-dimensional $\mathcal{C}$-networks. However a four-dimensional network $\tau(N)$ may be considered as a degenerate form of an n-dimensional one as follows. If $\bar{a} = (a_0, \ldots, a_{n-1})$ is an n-tuple of nodes of $\tau(N)$ then set

$$\tau(N)(\bar{a}) = \tau(N(a_0, a_1, a_2, a_3))$$

and this defines the corresponding n-dimensional $\mathcal{C}$ network $\tau(N)$. So any algorithm that solves the NSP over $\mathcal{C}$ will tell you if $\tau(N)$ is satisfiable or not.

Conversely, if $\tau(N)$ is satisfied in some representation of $\mathcal{C}$ then, by the condition in the theorem, $\tau(N)$ is satisfied in an representation Y of $\mathcal{C}$ of size bigger than five. Then, by lemma 24, there is a representation X of $\mathcal{D}_4$ and a one-dimensional, quantifier-free interpretation, based on τ, of X in Y. Thus N is satisfied in a representation X of $\mathcal{D}_4$.

Putting the two parts of the proof together, a 4-dimensional $\mathcal{D}_4$-network N is satisfiable in a representation of $\mathcal{D}_4$ if and only if the n-dimensional $\mathcal{C}$-network $\tau(N)$ is satisfiable in a representation of $\mathcal{C}$. Thus the network satisfaction problem for $\mathcal{D}_4$ reduces to that of $\mathcal{C}$. □

6.3 Complexity of NSP for Relation Algebras and CA_3

There is a correspondence between relation algebras and those three dimensional cylindric algebras that are generated by two dimensional elements [Mad91a]. We have seen that the following relation algebras have a network satisfaction problem with polynomial-time complexity: $\mathcal{A}_2, \mathcal{A}_3, \mathcal{P}$ and $\mathcal{L}$. On the other hand $\mathcal{M}$ and $\mathcal{I}$ have NP-complete network satisfaction problems. The complexity of the network satisfaction problem for the containment algebra $\mathcal{C}$ is not known. A necessary and sufficient set of conditions for a relation algebra to have a polynomial-time network satisfaction problem would advance the subject considerably. Although some progress has been made towards identifying the relation algebras with polynomial-time NSP an exact characterisation is still some way off.

In this section we give a further application of the work on interpretations by identifying a large class of relation algebras with NP-hard NSPs. This generalises some of the results in [Hir96]. If a two-dimensional translation of a relation algebra $\mathcal{A}$ in $\mathcal{B}$ is sufficiently expressive — i.e. it has the capability of expressing certain $L(\mathcal{B})$-formulas — then we can show that the network satisfaction problem for $\mathcal{A}$ is NP-hard.

THEOREM 26 *Let (τ, ∂) be a quantifier-free, two-dimensional, total translation from $\mathcal{A}$ to $\mathcal{B}$ such that*

1. *for each $k < \omega$ there is a representation Y of $\mathcal{B}$ such that $\partial(Y^2)$ contains a k-clique i.e.*

$$Y \models \exists y_0, \ldots, \exists y_{k-1} \bigwedge_{i \neq j < k} (\partial(y_i, y_j) \wedge y_i \neq y_j).$$

2. *there are elements $a_{ij} \in \mathcal{A}$ $(i, j < 2)$ such that*

$$T_{\mathcal{B}} \models \partial(x_0, x_1) \wedge \partial(y_0, y_1) \rightarrow (\tau(a_{ij})(x_0, x_1, y_0, y_1) \leftrightarrow Id_{\mathcal{B}}(x_i, y_j))$$

(or, roughly, $L(\mathcal{A})$ can express $x_i = y_j$).

Then the network satisfaction problem for $\mathcal{A}$ is NP-hard.

PROOF:

The argument is based on the proof of theorem 6. In that theorem the Hamiltonian circuit problem is reduced to the network satisfaction problem for $\mathcal{D}_4$. Given a graph G, a $\mathcal{D}_4$-network $N(G)$ is constructed such that G contains a Hamiltonian circuit if and only if $N(G)$ is satisfied in a (4-dimensional) representation of $\mathcal{D}_4$. Now for any graph G the network $N(G)$ has a special form: the nodes go in pairs n_0, n_1 and the only 4-tuples of $N(G)$ which are not labelled with 1 consist of two of these pairs. Also, n_0, n_1 are required to be distinct. In this proof we reduce the network satisfaction problem for $\mathcal{D}_4$ *restricted to this special form* to the (2-dimensional) network satisfaction problem for $\mathcal{A}$.

So let N be any $\mathcal{D}_4$-network with nodes $n_0^0, n_1^0, \ldots, n_0^{k-1}, n_1^{k-1}$ such that

- $N(\bar{n}) = 1$ unless $\bar{n} = (n_0^s, n_1^s, n_0^t, n_1^t)$ for some $s, t < k$

- $N(n_0^s, n_1^s, n_0^t, n_1^t) \leq (-d_{01})$ for any $s, t < k$ (i.e. n_0^s is not equal to n_1^s).

Recall that the network N corresponds to a $\mathcal{D}_4$-formula $\phi_N(x_0^s, x_1^s : s < k)$ where the free variables x_0^s, x_1^s correspond to the nodes n_0^s, n_1^s. Because N is a four-dimensional network, ϕ_N has the form

$$\bigwedge_{s,t<k} C_{st}(x_0^s, x_1^s, x_0^t, x_1^t)$$

i.e. a conjunction of four-variable, quantifier-free clauses C_{st} of the form shown above, where $C_{st} \in \mathcal{D}_4$. Every quantifier-free $\mathcal{D}_4$-formula is a boolean combination of diagonals.

Next define a $L(\mathcal{B})$-formula

$$\theta(N)(x_0^s, x_1^s : s < k) = \bigwedge_{s<k} \partial(x_0^s, x_1^s) \wedge \bigwedge_{s,t<k} C^*_{st}(x_0^s, x_1^s, x_0^t, x_1^t)$$

where C^*_{st} is obtained from C_{st} by replacing each diagonal d_{ij} by $Id_{\mathcal{B}}(y_i, y_j)$, where $y_0 = x_0^s, y_1 = x_1^s, y_2 = x_0^t, y_3 = x_1^t$. Because Y (in the statement of the theorem) is a representation of $\mathcal{B}$, it follows that N embeds in a representation of $\mathcal{D}_4$ if and only if $\theta(N)$ is satisfied in a representation of $\mathcal{B}$ — the argument is the same as in the proof of lemma 24.

Now we obtain an $L(\mathcal{A})$-formula $\psi(N)(x^s : s < k)$ with one free variable x^s for each pair of free variables x_0^s, x_1^s of $\theta(N)$. $\psi(N)$ is defined to be $\bigwedge_{s,t<k} C^+_{st}(x_s, x_t)$ where the clause $C^+_{st}(x_s, x_t)$ is obtained from $C^*_{st}((x_0^s, x_1^s, x_0^t, x_1^t)$ by replacing the atomic subformula $Id_{\mathcal{B}}(x_i^s, x_j^t)$ $(i, j < 2)$ by the $L(\mathcal{A})$-formula $a_{ij}(x^s, x^t)$. Let X be any representation of $\mathcal{A}$, let $x^s, x^t \in X$. Use left-totality to find a representation Y with an interpretation (τ, ∂, f) of X in Y. Let $y_0^s, y_1^s, y_0^t, y_1^t \in Y$ be such that $f(y_0^s, y_1^s) = x^s, f(y_0^t, y_1^t) = x^t$. By the definition of a_{ij}, a_{ij} holds at (x^s, x^t) in X if and only if $\partial(y_0^s, y_1^s) \wedge \partial(y_0^t, y_1^t) \wedge y_i^s = y_j^t$ holds in Y. Thus if $\psi(N)$ is satisfied in a representation of $\mathcal{A}$ then $\theta(N)$ holds in a representation of $\mathcal{B}$. Similarly, by right-totality, if $\theta(N)$ holds in a representation of $\mathcal{B}$ then $\psi(N)$ is satisfied in a representation of $\mathcal{A}$. In turn this holds if and only if N embeds in a representation of $\mathcal{D}_4$. Observe that each clause C^+_{st} of $\psi(N)$ is quantifier-free.

Finally, the $L(\mathcal{A})$-formula $\psi(N)(x^s : s < k)$ is a conjunction of two-variable, quantifier-free clauses $C^*_{st}(x_s, x_t)$. So each clause $C^*_{st}(x_s, x_t)$ must be equivalent to $\alpha_{st}(x_s, x_t)$ for some element $\alpha_{st} \in \mathcal{A}$. Thus the satisfiability of $\psi(N)$ is equivalent to the satisfiability of an $\mathcal{A}$-network M with nodes $m^s : s < k$ where $M(m^s, m^t) = \alpha_{st}$ for each pair $s, t < k$.

□

Note A generalisation of this theorem can be obtained by replacing the second condition (there are elements $a_{ij} \in \mathcal{A}$ ...) by the following condition. There are $L(\mathcal{A})$-formulas $\theta_{ij}(x, y)$ $(i, j < 2)$, definable by networks and such that

$$T_{\mathcal{B}} \models \partial(x_0, x_1) \wedge \partial(y_0, y_1) \rightarrow (\tau(\theta_{ij})(x_0, x_1, y_0, y_1) \leftrightarrow Id_{\mathcal{B}}(x_i, y_j))$$

COROLLARY 27 *The NSP for the seven atom relation algebra $\mathcal{B}_7$ is NP complete.*

PROOF:

The two atom algebra $\mathcal{A}_2$ and the seven atom relation algebra $\mathcal{B}_7$ are defined in the examples, section 3.2. The two-dimensional, total translation from $\mathcal{B}_7$ to the two-atom algebra $\mathcal{A}_2$ (page 27) meets the conditions of the theorem. Take any infinite representation of $\mathcal{A}_2$ to meet the first requirement in the theorem.

The $\mathcal{B}_7$-formulas a_{ij} $(i, j < 2)$ are defined as follows:

$$\begin{aligned} a_{11} &= \{Id_{\mathcal{A}_2}, \text{ Starts}\} \\ a_{12} &= \{\text{Swap }, \text{ Meets}^{\smile}\} \\ a_{21} &= \{\text{Swap }, \text{ Meets}\} \\ a_{22} &= \{Id_{\mathcal{A}_2}, \text{ Ends}\}. \end{aligned}$$

Thus, by the previous theorem, the complexity of the NSP over $\mathcal{B}_7$ is NP-hard.

To show that the complexity lies in NP, given a $\mathcal{B}_7$-network N, non-deterministically choose an atomic labeling L of N — i.e. for any nodes $p, q \in N$, $L(p, q)$ is an atom and $L(p, q) \in N(p, q)$. Such a labeling is consistent if and only if it's transitive closure is non-zero, which can be checked in cubic time. □

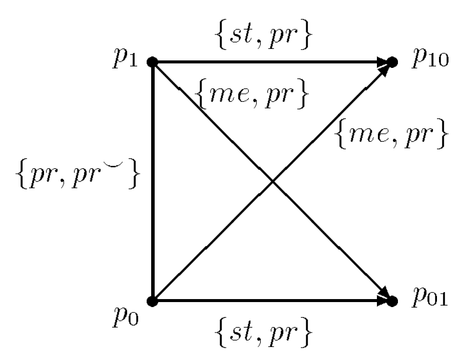
Next we give an application of this corollary, theorem 21 and lemma 20 to show that the NSP for the Allen interval algebra is NP-complete. In fact the original proof in [VK86] is preferable because of its simplicity and elegance, but the method here has wider application.

THEOREM 28 *The network satisfaction problem for the Allen interval algebra is NP-complete.*

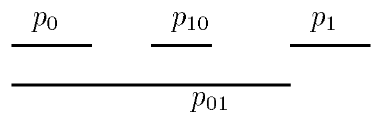
PROOF:

We use theorem 21 to show that the NSP for $\mathcal{B}_7$ reduces in polynomial time to the NSP for the Allen interval algebra $\mathcal{I}$. Because the two algebras have atoms with the same or similar names, we'll use the suffices $\mathcal{B}, \mathcal{I}$ to distinguish where necessary. Using lemma 20, we define a two-dimensional translation $(\tau, \partial) : \mathcal{B}_7 \to \mathcal{I}$ by defining each of the seven negated atoms, and the domain formula ∂ by networks. (In order to demonstrate that for each $b \in \mathcal{B}_7$, $\tau(b)$ is definable by a network $N(b)$, it suffices to meet this requirement for negated atoms only. This follows from lemma 20 because if $\tau(a)$ and $\tau(b)$ are definable by networks then so is $\tau(a \wedge b)$.) The domain formula $\partial(x_0, x_1)$ is given by $x_0\{\text{precedes}_{\mathcal{I}}, \text{precedes}_{\mathcal{I}}^{\smile}\}x_1$ and this is definable by a two-node network. The idea is that the start points of two disjoint intervals defines a pair of points in a representation of $\mathcal{B}_7$.

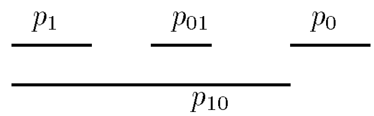
Let a be an atom of $\mathcal{B}_7$. We'll define $\tau(-a)(x_{p_0}, x_{p_1}, x_{q_0}, x_{q_1}) \in L(\mathcal{I})$ by an $\mathcal{I}$-network $N(-a)$ for each of the seven cases. In each case $N(-a)$ contains the nodes p_0, p_1, q_0, q_1 and $N(-a)(p_0, p_1) = N(-a)(q_0, q_1) = \{\text{precedes}_{\mathcal{I}}, \text{precedes}_{\mathcal{I}}^{\smile}\}$. For the last two cases only ($-Id_{\mathcal{B}}$ and $-\text{Swap}_{\mathcal{B}}$), we have additional nodes $p_{01}, p_{10}, q_{01}, q_{10}$.



Here we use the abbreviations pr, st, me for precedes, starts, meets respectively. For these two cases ($-Id_{\mathcal{B}}$ and $-\text{Swap}_{\mathcal{B}}$) we have $N(p_0, p_{01}) = N(q_0, q_{01}) = \{\text{starts }, \text{precedes}^{\smile}\}_{\mathcal{I}}$ and $N(p_1, p_{01}) = N(q_1, q_{01}) = \{\text{meets}, \text{precedes}\}_{\mathcal{I}}$. Similarly, $N(p_0, p_{10}) = N(q_0, q_{10}) = \{\text{meets}, \text{precedes}\}_{\mathcal{I}}$ and $N(p_1, p_{10}) = N(q_1, q_{10}) = \{\text{starts}, \text{precedes}^{\smile}\}_{\mathcal{I}}$. Note that if $p_0\{\text{precedes}\}p_1$ then p_{01} is constrained to be the interval that starts when p_0 starts and ends when p_1 starts.



On the other hand if $p_0\{\text{precedes}^{\smile}\}p_1$ then p_{10} starts when p_1 starts and ends when p_0 starts.



To define the network $N(-a)$ it remains to define the labels on edges joining a p to a q.

- $N(-\text{Meets}_{\mathcal{B}})$ is defined by letting $N(-\text{Meets}_{\mathcal{B}})(p_1, q_0) = -\{\text{starts}, \text{starts}^{\smile}, Id\}_{\mathcal{I}}$ and all other edges from a p to a q are labelled with $1_{\mathcal{I}}$.
- $\text{Meets}^{\smile}_{\mathcal{B}}, \text{Starts}_{\mathcal{B}}, \text{Ends}_{\mathcal{B}}$ have similar definitions (e.g. $N(-\text{Meets}_{\mathcal{B}})$ is obtained from $N(-\text{Meets}_{\mathcal{B}})$ by substituting q_1 for p_1 and p_0 for q_0.)
- $N(-\text{Disjoint}_{\mathcal{B}})$ is defined by $N(-\text{Disjoint}_{\mathcal{B}})(p_i, q_j) = \{\text{precedes}, \text{precedes}^{\smile}\}_{\mathcal{I}}$ for $i, j \in \{0, 1\}$.
- $N(-Id_{\mathcal{B}})$ is defined by letting $N(-Id_{\mathcal{B}})(p_{01}, q_{01}) = N(-Id_{\mathcal{B}})(p_{10}, q_{10}) = (-Id_{\mathcal{I}})$ and all others are labelled with $1_{\mathcal{I}}$.
- The network definition of $\tau(-\text{Swap})$ is similar but here $N(-\text{Swap})(p_{01}, q_{10}) = N(-\text{Swap})(p_{10}, q_{01}) = (-Id_{\mathcal{I}})$ (and all others are labelled with $1_{\mathcal{I}}$).

This translation extends, by lemma 20, to all elements of $\mathcal{B}_7$.

Now we know that every satisfiable $\mathcal{B}_7$-network embeds in the representation $\mathcal{B}(\mathbb{Q})$. This follows from corollary 4.

In order to meet the conditions of theorem 21 part 2 it remains to show that (τ, ∂) is left-total and that there is an interpretation, based on (τ, ∂), of $\mathcal{B}(\mathbb{Q})$ in a representation of $\mathcal{I}$. For left-totality, let Y be any representation of $\mathcal{I}$. By the previously mentioned result of [LM94] there is a base-isomorphism Ξ from Y to a representation Y^* of $\mathcal{I}$ whose domain consists of all ordered pairs from some dense linear order $(S, <)$. Let $\Xi(y) = (\Xi(y)^-, \Xi(y)^+)$. $\Xi(y)^- < \Xi(y)^+ \in S$. Then $\mathcal{B}(S)$ is a proper relation algebra isomorphic to $\mathcal{B}_7$, hence a representation of $\mathcal{B}_7$. We can define a surjection $f : \partial(Y^2) \to \mathcal{B}(S)$ by $f : (i, j) \mapsto (\Xi(i)^-, \Xi(j)^-)$ for disjoint intervals $i, j \in Y$. This defines a two-dimensional interpretation of $\mathcal{B}(S)$ in Y.

To interpret $\mathcal{B}(\mathbb{Q})$ in a representation Y of $\mathcal{I}$ let Y be the natural representation of $\mathcal{I}$ whose domain is ordered pairs of rational numbers. Define the co-ordinate map f by $f(i, j) = (i^-, j^-)$ where $i = (i^-, i^+)$ and $j = (j^-, j^+)$ are two disjoint intervals.

Hence, by theorem 21 part 2, the NSP for $\mathcal{B}_7$ reduces to that of $\mathcal{I}$.

□

7 Conclusion

In this paper we have analysed the complexity of the satisfaction problem and the network satisfaction problem for various relation algebras and cylindric algebras. The use of interpretations allows us to generalise known results to a wider range of algebras. Nevertheless the complexity map of these algebras is far from complete and we have indicated examples of relation algebras (like the containment algebra) where the complexity is still not known.

Of course for many interesting algebras the complexity is rather high. The problem of isolating tractable fragments of intractable algebras has been investigated for the particular case of the Allen interval algebra in [NB94, Lig94], but a general method needs to be found. Another approach to handling intractable algebras is to look for approximate solutions to the network satisfaction problem. The Allen propagation algorithm is a well-known example of this, but the game theoretic methods used in [HH97b] may well give more general results in this direction.

References

[AGN94] H Andréka, S Givant, and I Németi. Decision problems for equational theories of relation algebras. *Bulletin of Section of Logic*, (1), 1994. To appear.

[AH85a] J F Allen and P J Hayes. A commonsense theory of time. In A Joshi, editor, *Proceedings IJCAI*, pages 528–531. Morgan Kaufmann, 1985.

[AH85b] J F Allen and P J Hayes. Moments and points in an interval-based temporal logic. Technical Report TR180, Department of Computer Science, University of Rochester, 1985.

[AHU74] A Aho, J Hopcroft, and J Ullman. *The design and analysis of computer algorithms.* Addison-Wesley, 1974.

[AK83a] J F Allen and H Kautz. A model of naïve temporal reasoning. In R Moore J Hobbs, editor, *Contributions to Artificial Intelligence, Volume 1.* Ablex, Norwood, N.J., 1983.

[AK83b] J F Allen and J Koomen. Planning using a temporal world model. In *Proceedings of the 8th International Joint Conference on Artificial Intelligence*, volume 8, pages 711–714. Morgan Kaufmann, 1983.

[All81] J Allen. An interval-based representation of temporal knowledge. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence.* Morgan Kaufmann, August 1981.

[All83] J F Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, November 1983.

[All84] J F Allen. Towards a general theory of action and time. *Artificial Intelligence*, 23(2):123–154, 1984.

[AM94] H Andréka and R Maddux. Representations for small relation algebras. *Notre Dame Journal of Formal Logic*, 35(4):550–562, 1994.

[AMN91] H Andréka, J D Monk, and I Németi. *Algebraic Logic.* Amsterdam Colloq. Math. Soc. J. Bolyai. North-Holland, 1991. Conference Proceedings, Budapest, 1988.

[Com83] S Comer. A remark on chromatic polygroups. *Congr. Numer.*, pages 85–95, 1983.

[D9̈1] I Düntsch. Small integral relation algebras generated by a partial order. *Period. Math. Hungar.*, pages 129–138, 1991.

[DM87] T Dean and D McDermott. Temporal database management. *AI*, 32:1–55, 1987.

[DMP91] R Dechter, I Meiri, and J Pearl. Temporal constraint networks. *Artificial Intelligence*, 49:61–95, 1991.

[HH97a] R Hirsch and I Hodkinson. Complete representations in algebraic logic. *Journal of Symbolic Logic*, 1997. To appear.

[HH97b] R Hirsch and I Hodkinson. Step by step — building representations in algebraic logic. *Journal of Symbolic Logic*, 1997. To appear.

[Hir94a] R Hirsch. From points to intervals. *Journal of Applied Non-Classical Logics*, 4(1):7–27, 1994.

[Hir94b] R Hirsch. *Relation Algebras of Intervals.* PhD thesis, Imperial College, 1994.

[Hir95] R Hirsch. Intractability in the Allen & Koomen planner. *Computational Intelligence Journal*, 11(4), November 1995.

[Hir96] R Hirsch. Relation algebras of intervals. *Artificial Intelligence Journal*, 83:1–29, 1996.

[HMT71] L Henkin, J D Monk, and A Tarski. *Cylindric Algebras Part I.* North-Holland, 1971.

[HMT85] L Henkin, J D Monk, and A Tarski. *Cylindric Algebras Part II.* North-Holland, 1985.

[Hod95] W Hodges. *Model theory.* Encylopedia of mathematics and its applications. Cambridge University Press, 1995. second reprint.

[J8́2] B Jónsson. Varieties of relation algebras. *Algebra Universalis*, 15:273 – 298, 1982.

[J9́1] B Jónsson. The theory of binary relations. In Andréka *et. al.,* [AMN91], pages 245 – 292. Conference Proceedings, Budapest, 1988.

[JT48] B Jónsson and A Tarski. Representation problems for relation algebras. *Bulletin of the American Mathematical Society*, 54, 1948.

[KL91] H Kautz and P Ladkin. Integrating metric and qualitative temporal reasoning. In *Proceedings of the ninth national conference on artificial intelligence, Angeim, California*, pages 241–246, 1991.

[Koo89] J Koomen. Localising temporal constraint propagation. In Brachman, Levesque, and Reiter, editors, *Proceedings of the first conference on principles of Knowledge Representation and Reasoning*, pages 198–202. Morgan Kaufmann, 1989.

[KV86] H Kautz and M Vilain. Constraint propagation algorithms for temporal reasoning. In *Proceedings of the fifth national conference on artificial intelligence*, volume 86, pages 377–382. MIT, 1986.

[Lig94] G Ligozat. Tractable relations in temporal reasoning: pre-convex relations. In Neumman, editor, *ECAI-94, proceedings on spatial and temporal reasoning*, Amsterdam, August 1994. John Wiley and Sons Ltc.

[LM88] P Ladkin and R Maddux. On binary constraint networks. Technical Report KES.U.88.8, The Kestrel Institute, 1988.

[LM93] P Ladkin and R Maddux. On binary constraint problems. *Journal of the American Mathematical Society*, 1993.

[LM94] P Ladkin and R Maddux. On binary constraint problems. *Journal of the Association of Computing Machinery*, 41:435 – 469, 1994.

[LR93] P Ladkin and A Reinefeld. A symbolic approach to interval constraint problems. *Artificial Intelligence and Symbolic Mathematical Computing*, 737:65–84, 1993.

[Lyn50] R Lyndon. The representation of relational algebras. *Annals of Mathematics*, 51(3):707–729, 1950.

[Mad91a] R Maddux. Introductory course on relation algebras, finite-dimensional cylindric algebras, and their interconnections. In Andréka *et. al.,* [AMN91], pages 361–392. Conference Proceedings, Budapest, 1988.

[Mad91b] R Maddux. The origin of relation algebras in the development and axiomatization of the calculus of relations. *Studia Logica*, 3/4, 1991.

[McK66] R McKenzie. *The representation of relation algebras.* PhD thesis, University of Colorado at Boulder, 1966.

[Mon64] J D Monk. On representable relation algebras. *Michigan Mathematics Journal*, 11:207–210, 1964.

[NB94] Bernhard Nebel and Hans-Jürgen Bürckert. Reasoning about temporal relations: A maximal tractable subclass of Allen's interval algebra. In *Proceedings of the 12th National Conference of the American Association for Artificial Intelligence*, pages 356–361, Seattle, WA, July 1994. MIT.

[Pel88] R Pelavin. *A formal approach to planning with concurrent actions and external events.* PhD thesis, University of Rochester, 1988.

[Sto36] M Stone. The theory of representations for boolean algebras. *Transactions of the American Mathematical Society*, 40:37–111, 1936.

[TG87] A Tarski and S R Givant. *A Formalization of Set Theory Without Variables.* Number 41 in Colloquium Publications in Mathematics. American Mathematical Society, Providence, Rhode Island, 1987.

[vEB94] P van Emde Boas. Machine models and simulations. In van Leeuwen [vL94], pages 1–61.

[VK86] M Villain and H Kautz. Constraint propagation algorithms for temporal reasoning. In *Proceedings of the fifth AAAI*, pages 377–382, 1986.

[VKvB89] M Villain, H Kautz, and P van Beek. Constraint propagation algorithms for temporal reasoning — a revised report. *Readings in Qualitative Reasoning about Physical Systems*, pages 373–381, 1989.

[vL94] J van Leeuwen. *Handbook of theoretical computer science.* MIT, 1994.