A Finite Relation Algebra with Undecidable Network Satisfaction Problem

ROBIN HIRSCH, *Department of Computer Science, University College London, Gower Street, London WC1, UK.*
E-mail: R.Hirsch@cs.ucl.ac.uk.
URL: http:www.cs.ucl.ac.uk/staff/R.Hirsch/

Abstract

We define a finite relation algebra and show that the network satisfaction problem is undecidable for this algebra[1].

Keywords: Network satisfaction problem, relation algebra, undecidability, tiling

1 Notation and Definitions

Let $\mathcal{A} = (A, +, -, 0, 1, 1', \breve{}\,, ;)$ be a relation algebra (see [JT52] for the original axiomatisation or [Mad91] for an introduction to relation algebra).

- An *atom* a of $\mathcal{A}$ is a minimal, non-zero element. $At(\mathcal{A})$ denotes the set of all atoms of $\mathcal{A}$. $\mathcal{A}$ is *atomic* if for all non-zero $a \in \mathcal{A}$ there exists $\alpha \in At(\mathcal{A})$ with $\alpha \leq a$. In the following we assume $\mathcal{A}$ is atomic.
- The *set of forbidden triples* of the atomic relation algebra $\mathcal{A}$ is the set of all $(\alpha, \beta, \gamma) \in {}^3At(\mathcal{A})$ such that $\breve{\gamma} \cdot \alpha;\beta = 0$. Forb denotes the set of forbidden triples of $\mathcal{A}$.
- The set of forbidden triples defines composition in an atomic relation algebra, by

$$a;b = \Sigma\{\gamma \in At(\mathcal{A}) : \exists \alpha, \beta \in At(\mathcal{A}),\ \alpha \leq a,\ \beta \leq b,\ \&(\alpha, \beta, \breve{\gamma}) \notin \mathsf{Forb}\}$$

- If (α, β, γ) is a forbidden triple then so are the six Peircean transforms: $(\alpha, \beta, \gamma), (\beta, \gamma, \alpha), (\gamma, \alpha, \beta), (\breve{\alpha}, \breve{\gamma}, \breve{\beta}), (\breve{\beta}, \breve{\alpha}, \breve{\gamma}), (\breve{\gamma}, \breve{\beta}, \breve{\alpha})$ (this follows from the axioms for relation algebras).
- A *network*[2] N over $\mathcal{A}$ is a map from $nodes(N) \times nodes(N)$ into $\mathcal{A}$, for some set $nodes(N)$.
- A network N is *3-consistent* or *path-consistent* if for all $l, m, n \in nodes(N)$ we have $N(l, n) \leq N(l, m); N(m, n)$ and $N(l, l) \leq 1'$. It is easy to show that path-consistency implies $N(l, m) = N(m, l)^{\smile}$, for all $l, m \in nodes(N)$.

---

[1]Partially supported by UK EPSRC grant GR/L85961. Thanks to Ian Hodkinson, Sean Holden and Roger Maddux for their contributions to this result. Thanks especially to the anonymous referee for pointing out several errors and improving the paper in many ways.

[2]Here we follow the temporal reasoning literature and impose no constraints on N. Elsewhere we include in the definition of N certain consistency requirements.

- If $N(m,n)$ is an atom of $\mathcal{A}$ for all $m,n \in nodes(N)$ then the network N is called an *atomic network*.
- A *representation* h of $\mathcal{A}$ maps each element of $\mathcal{A}$ to a binary relation over some domain X such that h is a boolean isomorphism and

$$\begin{aligned} h(1') &= \{(x,x) : x \in X\} \\ h(\breve{a}) &= \{(x,y) : (y,x) \in h(a)\} \\ h(a;b) &= \{(x,y) : \exists z \in X,\ (x,z) \in h(a) \wedge (z,y) \in h(b)\} \end{aligned}$$

- Let h be a representation of $\mathcal{A}$ over the domain X and let N be a network. A map $'$ from $nodes(N)$ into X is called an *embedding of N into h* if

$$(m', n') \in h(N(m,n))$$

  for all $m, n \in nodes(N)$.
- The *network satisfaction problem* (NSP) over $\mathcal{A}$ is to determine for an arbitrary network N over $\mathcal{A}$ whether there is a representation and an embedding of N into that representation.

2 Background

A number of relation algebras have been used for temporal reasoning. There are cases where the *network satisfaction problem* (NSP) is tractable, e.g. this is the case for the three atom *point algebra* [DMP91] and the *left-linear algebra* [Com83, D91, AGN94]. But typically the NSP is **NP**-complete as, for example, with the Allen interval algebra [All83, All84] (**NP**-completeness proved in [VK86, theorem 2]). To show that the NSP for the Allen interval algebra is in **NP** consider the following non-deterministic algorithm. For each edge of a given network, non-deterministically pick one atom below the element that labels that edge. If the resulting atomic network M is 3-consistent (and this can be checked in cubic time) then the original network is satisfiable (this follows from results in [LM94]). If each possible set of choices leads to an atomic network that fails 3-consistency then the original network is unsatisfiable. This non-deterministic algorithm runs in cubic time and solves the NSP for the Allen interval algebra and works also for many other relation algebras.

However, it is not true for all relation algebras that a 3-consistent atomic network is necessarily satisfiable. An example where this can fail is the *pentagonal algebra* [Mad91] with three self-converse atoms $1', e, d$. Composition is defined by listing the forbidden triples of atoms (see above). The forbidden triples consist of all Peircean transforms of $(1', x, y)$ for $x \neq y \in \{1', e, d\}$, (e,e,e) and (d,d,d). For this algebra it is possible to construct a 3-consistent atomic network where the network in not satisfiable in any representation of the algebra. See figure 1. For the pentagonal algebra it turns out to be the case that the NSP is in **NP**, but the question is posed: can the complexity of the NSP be worse than **NP** and, if so, how bad can the complexity be?

In this paper we show that the NSP is undecidable for a certain finite relation algebra. This is done by reducing an undecidable tiling problem to it. The construction of the relation algebra is the same construction as we gave in [HH99] to show that

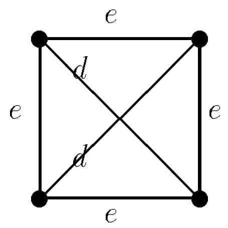

FIG. 1: A 3-consistent but unsatisfiable atomic network over the pentagonal algebra [Mad91, page 389]

the problem of determining whether a finite relation algebra is representable or not is an undecidable one[3]. In this paper we give the construction again and prove the easy half of the main theorem. We hope this gives some insight as to why the construction works. We omit the harder half of the proof but refer to the corresponding proof in [HH99].

No worse complexity is possible because for any fixed, finite relation algebra $\mathcal{A}$ the unsatisfiable finite networks over $\mathcal{A}$ are recursively enumerable. This follows from results in [HH97, section 9.1].

3 Tilings

Let τ be a fixed, finite set of tiles with horizontal adjacency $\mathcal{H} \subseteq \tau \times \tau$ and vertical adjacency $\mathcal{V} \subseteq \tau \times \tau$. In the following, we may sometimes simply write τ for a set of tiles and take the adjacencies to be given, provided this is unambiguous. An instance of the decision problem $P(\tau)$ is a non-empty, finite sequence $S(0,0), S(1,0), \ldots, S(n,0) \in \tau$ such that $(S(i,0), S(i+1,0)) \in \mathcal{H}$, for each $i < n$. Such an instance is a yes-instance if it is possible to extend this finite, one-row fragment into a tiling of the whole plane $S(i,j) : i,j \in \mathbb{Z}$ where $(S(i,j), S(i+1,j)) \in \mathcal{H}$ and $(S(i,j), S(i,j+1)) \in \mathcal{V}$ for $i,j \in \mathbb{Z}$, and it is a no-instance if it is impossible to extend to such a tiling.

LEMMA 3.1
There exists a finite set of tiles τ such that $P(\tau)$ is undecidable.

PROOF. Let U be any determinisitc Turing machine (with a two-way infinite tape) that recognizes a recursively enumerable but not recursive language: such machines are known to exist. So the problem of deciding whether U halts or not, starting on an arbitrary string w in the input alphabet of U, is undecidable. There are a number of ways of coding up U as a finite set of tiles and adjacencies τ so that successive rows of any tiling that might exist represent the configurations of U at successive times. Let $\delta : Q \times \Sigma \to Q \times (\Sigma \cup \{\mathcal{L}, \mathcal{R}\})$ be the transition function of U, where Q is the set of states, Σ is the alphabet of U and $\mathcal{L}$ and $\mathcal{R}$ represent an instruction to move left or right respectively. For an example of such a coding, let $b \in \Sigma$ be the blank symbol and let $\Sigma' = \Sigma \cup \{b_l, b_r\}$ (we'll use these extra symbols for blanks on the left and

[3] Indeed the same construction can be used to show for any $n \geq 5$ that the problem of deciding whether a finite relation algebra is a subalgebra of some relation algebras derived from any cylindric algebra of dimension n is undecidable.

right, respectively, of an input string). We extend the transition function δ to δ' by letting $\delta'(q, b_l) = \delta'(q, b_r) = \delta(q, b)$ for each $q \in Q$. Let τ include a tile $T(q, s, x)$ for each $q \in Q$, $s \in \Sigma'$, $x \in \{L, C, R, O\}$ plus one additional tile Y. $x = C$ is intended to denote that the tape head is in the position of that tile, $x = L$ (or R) is used to denote that the tape head is about to move left (or right) onto that tile and $x = O$ is used otherwise. Y will be used to extend any tiling of the upper half-plane to a tiling of the whole plane. Next we define horizontal and vertical adjacencies $\mathcal{H}, \mathcal{V} \subseteq \tau \times \tau$. Let $\mathcal{H} =$

$$\begin{array}{rcl}
\{(T(q,s,x), T(q,s',x')) & : & q, q' \in Q,\ s, s' \in \Sigma',\ x, x' \in \{L, C, R, O\}, \\
 & \wedge & \text{if } x' = R \text{ then } x = C \text{ and } \delta'(q,s) = \mathcal{R} \\
 & \wedge & \text{if } x = L \text{ then } x' = C \text{ and } \delta'(q,s') = \mathcal{L} \\
 & \wedge & \text{if } s' = b_l \text{ then } s = b_l \\
 & \wedge & \text{if } s = b_r \text{ then } s' = b_r\} \\
\cup\{(Y,Y)\} & &
\end{array}$$

and $\mathcal{V} =$

$$\begin{array}{rcl}
\{(T(q,s,x), T(q',s',x')) & : & \text{if } x \neq C \text{ then } s = s' \\
 & \wedge & \text{if } x = C \text{ then either } \delta'(q,s) = (q',s') \text{ and } x' = C \\
 & & \qquad \text{or } \delta'(q,x) = (q',L) \text{ or } (q',R) \text{ and } s = s' \\
 & \wedge & x' = C \text{ iff either } x = C \text{ and } \delta'(q,s) = (q',s') \\
 & & \qquad \text{or } x = L \text{ or } x = R \\
\} & &
\end{array}$$

$$\cup\ \{(Y,Y), (Y, T(q,s,x)) : q \in Q,\ s \in \Sigma,\ x \in \{L, C, R, O\}\}$$

We say that T' can go on the right of T if $(T, T') \in \mathcal{H}$ and that T' can go above T if $(T, T') \in \mathcal{V}$.

Now we reduce the undecidable word recognition problem for U to $P(\tau)$. Let $w = (w_1, \ldots, w_n)$ (some n) be a string in the alphabet Σ and let U's start state be q_0. Construct an instance $S = S(w)$ of $P(\tau)$ by letting $S(0,0) = (q_0, b_l, x_0)$, $S(0, n+1) = (q_0, b_r, x_{n+1})$ and $S(0,i) = T(q_0, w_i, x_i)$ for each i with $1 \leq i \leq n$ where

- $x_1 = C$
- $x_i = O$ for $3 \leq i \leq n+1$
- $x_2 = R$ if $\delta'(q_0, w_1) = (q, R)$ (any $q \in Q$) and $x_2 = O$ otherwise
- $x_0 = L$ if $\delta'(q_0, w_1) = (q, L)$ (any q) and $x_0 = O$ otherwise.

This gives an instance of $P(\tau)$. To show that this is a correct reduction, suppose first that $S(w)$ is a yes-instance of $P(\tau)$, i.e. it extends to a tiling $S(i,j) : i, j \in \mathbb{Z}$ of the plane. Since $S(0,0) = (q_0, b_l, x_0)$ the definition of horizontal adjacency shows that $S(i, 0) = (q_0, b_l, x_i)$ (some x_i) for each $i < 0$ and similarly $S(i, 0) = (q_0, b_r, x_i)$ for each $i > n+1$. Thus row 0 represents the initial configuration of U at time 0 with the tape head at position 1. Using the vertical adjacency we see that for $j \geq 0$, the j'th row $S(i,j) : i \in \mathbb{Z}$ represents the configuration of U at time j. Since the tiling goes on forever, this means that U will run forever on input w, so w is a no-instance of U. (It is slightly irritating that a yes-instance of the tiling problem corresponds to a no-instance of the recognition problem for U, but this can't be helped.) Conversely, if U runs forever on input w then for $j \geq 0$ let the tape contents at time j be $v(i,j) : i \in \mathbb{Z}$ and let the state be q_j. We construct a tiling of the plane by letting $S(i,j) = T(q_j, v(i,j), x_{ij})$ for $j \geq 0$, where $x_{ij} = C$ if the tape head is in position i at time j; $x_{ij} = R$ if the tape head is in position $i-1$ at time j and in position i at time $j+1$; $x_{ij} = L$ if the tape head is in position $i+1$ at time j and in position i at time $j+1$; and $x_{ij} = O$ otherwise. For $j < 0$ we let $S(i,j) = Y$. This gives a tiling of the plane and shows that $S(w)$ is a yes-instance. ∎

We now modify the tiles τ for technical reasons involved in the proof of the second part of theorem 4.1. These modifications are needed in order to apply [HH99, theorem 4]. Let τ be a set of tiles with adjacencies $\mathcal{H}, \mathcal{V}$, as above. Define a modified set of tiles τ' from τ by

$$\tau' = \{T \in \tau : \text{there is a } (\tau) \text{ tiling of the plane with } T \text{ at } (0,0)\} \cup \{Z\}$$

for some new tile $Z \notin \tau$. For the adjacencies, $\mathcal{H}', \mathcal{V}'$, if $S, T \in \tau \cap \tau'$ we let $(S,T) \in \mathcal{H}' \iff (S,T) \in \mathcal{H}$ and $(S,T) \in \mathcal{V}' \iff (S,T) \in \mathcal{V}$ and for the extra tile we let $(Z,Z) \in \mathcal{H}' \cap \mathcal{V}'$ but no other tiles are adjacent to Z. Observe that the new tile Z can tile the plane on its own but not in combination with any other tile.

Lemma 3.2
Let τ be a set of tiles such that $P(\tau)$ is undecidable. Let τ' be defined from τ as above. Then

1. $P(\tau')$ is undecidable
2. for each tile $T \in \tau'$ there is a tiling of the plane with T placed at $(0,0)$ and
3. there is a tile $Z \in \tau'$ which can tile the plane on its own but cannot be adjacent to any other tile.

These are the exact conditions required for the application of [HH99, theorem 4].

Proof. The last two parts follow straight from the definition of τ'. For the first part, suppose for contradiction that $P(\tau')$ were decidable. Then a decision algorithm for $P(\tau)$ can be obtained, contrary to the condition of the lemma. For the algorithm, take any instance $\bar{S}$ of $P(\tau)$. If $\bar{S}$ contains any tile T not in τ' then there is no tiling of the plane with T at $(0,0)$ hence no tiling of the plane containing T at all. So $\bar{S}$ is a no-instance. Otherwise, if every tile in $\bar{S}$ belongs to τ', then use the assumed decision algorithm for $P(\tau')$ to decide if $\bar{S}$ is a yes-instance or a no-instance of $P(\tau)$. ∎

4 A relation algebra with an undecidable NSP

Theorem 4.1
There is a finite relation algebra $\mathcal{A}$ such that the NSP over $\mathcal{A}$ is undecidable.

Proof. The proof works by reducing the problem $P(\tau')$ of lemma 3.2 to the NSP for a certain finite relation algebra $\mathcal{A}(\tau')$, defined in [HH99]. If τ' has k tiles then $\mathcal{A}(\tau')$

has $2k + 28$ atoms. They are:

$$\begin{array}{ll} e_0, w_0 & \\ e_i, w_i, +1_i, -1_i & i = 1, 2 \\ g_{0i}, u_{0i}, v_{0i}, w_{0i} & i = 1, 2 \\ w_{12}, T_{12} & T \in \tau' \end{array}$$

plus the converses of all atoms with two distinct suffices, viz.

$$\begin{array}{ll} \breve{g_{0i}} = g_{i0},\ \breve{u_{0i}} = u_{i0},\ \breve{v_{0i}} = v_{i0},\ \breve{w_{0i}} = w_{i0} & i = 1, 2 \\ \breve{w_{12}} = w_{21},\ \breve{T_{12}} = T_{21} & T \in \tau' \end{array}$$

The identity is given by $1' = e_0 + e_1 + e_2$. Converse is defined on atoms with two distinct suffices by reversing the order of the suffices. All other atoms are self-converse except $\breve{+1_i} = -1_i$ and $\breve{-1_i} = +1_i$ $(i = 1, 2)$. Composition is defined by listing the forbidden triples of atoms. Any triple of atoms where the subscripts do not match is forbidden, so $(x_{i,j}, y_{j',l'}, z_{l',i'})$ is forbidden unless $i = i', j = j'$ and $l = l'$. Here $x_{i,j}, y_{j',l'}, z_{l',i'}$ stand for any atoms with the appropriate subscripts: we handle the case of atoms with a single subscript by treating it as a repeated subscript, e.g. $e_0 = e_{00}$. Secondly, any of the six Peircean transforms of the triple (e_i, b, c) is forbidden if $b \neq \breve{c}$. Finally, all Peircean transforms of

$$(g_{10}, g_{02}, w_{21}) \tag{4.1}$$

$$(T_{12}, S_{21}, +1_1) \quad \text{any } i, j < k, \text{ if } (S, T) \notin \mathcal{H}' \tag{4.2}$$

$$(u_{10}, g_{02}, T_{21}) \quad \text{any } T \in \tau' \setminus \{Z\} \tag{4.3}$$

$$(v_{10}, g_{01}, +1_1), (v_{10}, g_{01}, -1_1) \tag{4.4}$$

are forbidden. There are three dual rules for forbidden triples, obtained from rules (4.2), (4.3) and (4.4) by swapping the subscripts 1 and 2 throughout and replacing $\mathcal{H}'$ by $\mathcal{V}'$. All other triples of atoms are allowed.

Now take an arbitrary instance $\bar{S} = (S(0,0), S(1,0), \ldots S(n,0))$ of $P(\tau')$. We construct an atomic network $N = N(\bar{S})$ over $\mathcal{A}(\tau')$ with $n + 3$ nodes $z, y_0, x_0, \ldots, x_n$ in such a way that $\bar{S}$ is a yes-instance of $P(\tau')$ if and only if N is a satisfiable atomic network. The labelling of N is given by $N(z, z) = e_0$, $N(y_0, y_0) = e_2$, $N(x_i, x_i) = e_1$, $N(z, y_0) = g_{02}$, $N(z, x_i) = g_{01}$ (each $i \leq n$), $N(x_i, x_{i+1}) = +1_1$ (each $i < n$), $N(x_i, x_j) = w_1$ (all $i, j \leq n$ with $|i - j| > 1$) and $N(x_i, y_0) = S(i, 0)_{12}$ (each $i \leq n$). See figure 2.

Claim 1 If N is satisfiable then $\bar{S}$ is a yes-instance of $P(\tau')$.

Proof of claim 1. Let $h : \mathcal{A}(\tau') \to \wp(X \times X)$ be a representation of $\mathcal{A}(\tau')$ over some domain X such that $' : nodes(N) \mapsto X$ is an embedding of the network N into the representation. So, for $m, n \in nodes(N)$ we have $(m', n') \in h(N(m, n))$. We have $N(z, y_0) = g_{02} \leq g_{02}; -1_2$, so $(z', y'_0) \in h(g_{02}; (-1_2))$. Hence there is a point $y'_1 \in X$ such that $(z', y'_1) \in h(g_{02})$ and $(y'_1, y'_0) \in h(-1_2)$, or equivalently $(y'_0, y'_1) \in h(+1_2)$. Similarly we can find points $y'_2, y'_3, \ldots \in X$ and $x'_{n+1}, x'_{n+2}, \ldots \in X$ such that $(z', y'_i) \in h(g_{02})$, $(y'_i, y'_{i+1}) \in h(+1_2)$ for $i = 0, 1, 2, \ldots$ and $(z', x'_i) \in h(g_{01})$, $(x'_i, x'_{i+1}) \in h(+1_1)$ for $i = 0, 1, 2, \ldots$. Extending the sequences downward $x_{-1}, x_{-2}, \ldots,$ $y_{-1}, y_{-2}, \ldots$ is entirely similar. See figure 3.

For each $i, j \in \mathbb{Z}$ consider the triangle (z', x'_i, y'_j). Since $(x'_i, z') \in h(g_{10})$ and $(z', y'_j) \in h(g_{02})$ it follows that $(x'_i, y'_j) \in h(g_{10}; g_{02})$. Since $\mathcal{A}(\tau')$ is a finite algebra,

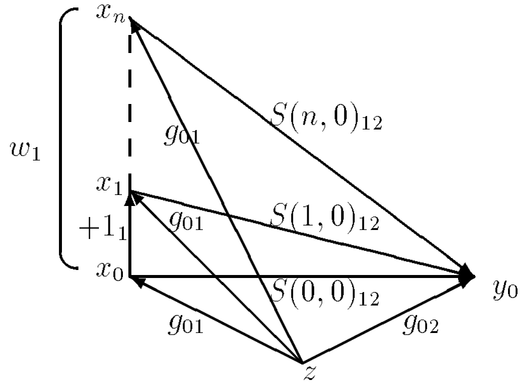

FIG. 2. The atomic networks $N(\bar{S})$.

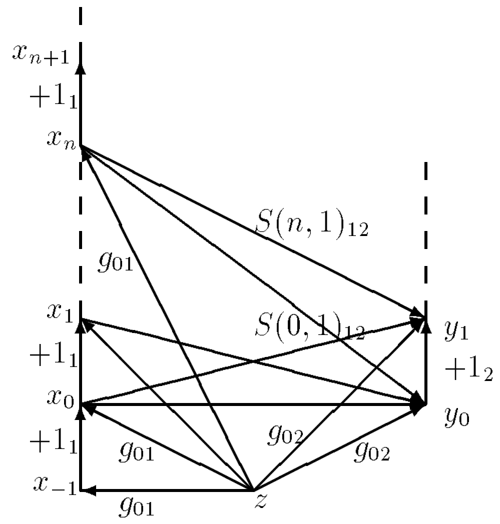

FIG. 3. Extending $N(\bar{S})$

there must be an atom $a(i,j) \leq g_{10}; g_{02}$ such that $(x'_i, y'_j) \in h(a(i,j))$. By the rule of matching subscripts, $a(i,j)$ must have subscripts (12). $a(i,j) = w_{12}$ is impossible by rule (4.1), hence $a(i,j) = S(i,j)_{12}$ for some tile $S(i,j) \in \tau'$. Thus the network $N(\bar{S})$ can be extended as in figure 3.

It remains to show that the tiles $S(i,j) : i,j \in \mathbb{Z}$ form a tiling of the plane. By considering the triangle (x'_i, x'_{i+1}, y'_j) and rule (4.2) we see that $S((i,j), S(i+1,j)) \in \mathcal{H}'$ and similarly $(S(i,j), S(i,j+1)) \in \mathcal{V}'$ for $i,j \in \mathbb{Z}$. Hence we have a tiling of the plane extending $\bar{S}$, so $\bar{S}$ is a yes-instance.

Claim 2. If $\bar{S}$ is a yes-instance of $P(\tau')$ then $N = N(\bar{S})$ is satisfiable.

The proof of claim 2 is much more complicated and makes use of the new tile Z, the atoms u_{0i}, v_{0i} $(i = 1,2)$ and rules (4.3) and (4.4). The reader is referred to [HH99, theorem 4]. If $\bar{S}$ is a yes-instance of $P(\tau')$ then it is possible to extend the finite fragment $\bar{S}$ to a tiling of the whole plane $\mathbb{Z} \times \mathbb{Z}$. Now (by definition of τ') if $T \in \tau'$ there is a tiling of the plane with T at $(0,0)$ and there is a special tile $Z \in \tau'$ which can tile the plane on its own, but not in combination with any of the other tiles. These are the conditions required in [HH99, theorem 4]. The theorem tells us that the second player ($\exists$) has a winning strategy in a certain game $G_\omega(\mathcal{A}(\tau'))$ which suffices to prove that $\mathcal{A}(\tau')$ is representable, but we can prove more than this.

Here we consider instead the game $G_\omega(N, \mathcal{A}(\tau'))$ which is identical to $G_\omega(\mathcal{A}(\tau'))$ except that the play starts from the initial network N. Using the terminology of [HH99], we let the edge (x_0, y_0) and all the edges labelled by $g_{01}, g_{02}, +1_1$ belong to $\forall$ along with the converses of all these edges (see figure 2). All other edges of N belong to $\exists$. Then the definition of $\exists$'s strategy and the proof that this is a winning one go through unaltered. This suffices to provide a representation of $\mathcal{A}(\tau')$ in which N embeds. Thus N is satisfiable.

Hence the undecidable tiling problem $P(\tau')$ reduces to the network satisfaction problem over $\mathcal{A}(\tau')$. We conclude that the latter is also undecidable.

∎

References

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

[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.

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

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

[D91] I Düntsch. Small integral relation algebras generated by a partial order. *Period. Math. Hungar.*, pages 129–138, 1991.

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

[HH97] R Hirsch and I Hodkinson. Step by step — building representations in algebraic logic. *Journal of Symbolic Logic*, 62(1):225–279, March 1997.

[HH99] R Hirsch and I Hodkinson. Representability is not decidable for finite relation algebras. *Transactions of the American Mathematical Society*, 1999. To appear.

[JT52] B Jónsson and A Tarski. Boolean algebras with operators ii. *American Journal of Mathematics*, 74:127 – 162, 1952.

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

[Mad91] 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.

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

Received 28/May/1999