COMPUTING WHAT A RELATION FAILS TO EXPRESS: KERNEL THEOREMS FOR TRANSITION AUTOMATA

CHRYSTOPHER L. NEHANIV
SCHOOL OF COMPUTER SCIENCE AND ENGINEERING
THE UNIVERSITY OF AIZU, 985-80 JAPAN

ABSTRACT. We define the kernel of a relational morphism of finite or infinite, faithful or non-faithful state-transition automata. We prove the Covering Lemma for these automata, and a characterization of embedding in the wreath product in terms of this kernel.

1. INTRODUCTION

For groups, the kernel K of a morphism $\varphi : G \longrightarrow H$ is again a group, and G embeds in the wreath product $K \circ H$, ("undoing" the morphism in a Lagrange coordinates). For semigroups and automata, the situation is analogous but more complex. In 1987, B. Tilson published his seminal Derived Category Theorem paper [4] which defines the kernel of a monoid (or semigroup) morphism. This kernel of a morphism turns out to be a small category, rather than a monoid. Tilson's Covering Lemma (the Derived Category Theorem) says that wreathing a divisor[1] of the kernel to the image allows one to "undo" the morphism (up to division). In 1989, Rhodes and Tilson [3] gave a closely related kernel and theorems for the two-sided wreath product (the *block product*).

The desirability for a corresponding theorem for wreath products of transformation semigroups was mentioned in [1]. And in [2] the author has given a tighter analogue to Tilson's Covering Lemma for morphisms – and more generally relational morphisms – in the setting of (possibly non-faithful, finite or infinite) transformation semigroups, and has described an embedding condition for wreath products of transformation semigroups in terms of this kernel, and has also given applications including a short proof of the Krohn-Rhodes Theorem, wreath product coordinates on the natural representation of Teissier semigroups (right-simple idempotent free), and very tight Lagrange coordinates on transformation groups.

This paper proves analogues of the kernel theorems of [2] in the setting of transition automata, a setting in which their proofs undergo some simplication. We define a new kernel for relational morphisms of transition automata. By considering transition automata, we are able to prove the Covering Lemma:

Author's e-mail address: `nehaniv@u-aizu.ac.jp`

‡This research was supported in part by the Algebraic Engineering of Computer Arithmetic Project and FAST (Framework for Advanced Software Techniques) Project, University of Aizu, Fukushima Prefecture.

[1] In Tilson's sense of division, which allows a small category to divide a monoid.

In the case of a faithful transition automaton (X, Σ): Given a [relational] morphism R from (X, Σ) to (Y, Ξ): The transition automaton (X, Σ) is emulated (i.e. covered by, i.e. divides) $(Z, \Upsilon) \circ (Y, \Xi)$ inducing R if and only (Z, Υ) "computes" the kernel of the [relational] morphism. For non-faithful (X, Σ) the "if" direction is also proved. (See below for precise definitions.)

We also prove an Embedding Computation Theorem for transition automata, which describes how to undo a surjective morphism to obtain an embedding of the morphism's domain into a wreath product with the morphism's target. This is a characterization of embedding in a wreath product in terms of the kernel, and thus has a stronger form than the corresponding theorem for transformation semigroups [2].

These results apply to both finite and infinite transition automata.

2. Definitions

A transition automaton (X, Σ) consists of a set X of *states* and a *inputs* Σ which act on X in a fixed way

$$\lambda : X \times \Sigma \to X.$$

That is, if the state is currently $x \in X$, and $s \in \Sigma$ is input, then the resulting state is $\lambda(x, s)$. We also denote the latter as $x \cdot s$, notationally surpressing the *transition function* λ. The *wreath product* of transition automata (X, Σ) and (Y, Ξ) is a transition automaton with states $X \times Y$ and inputs $\Sigma^Y \times \Xi$: an input f consists of an input $t \in \Xi$ and a function $\overline{f}$ from Y to Σ. A transition from a state (x,y) to the the next state upon inputing f is computed by

$$(x, y)f = (x \cdot \overline{f}(y), y \cdot t).$$

The wreath product is a "generic" cascade product: any cascade of two automata embeds into their wreath product.

A transtion automaton is *faithful* if $x \cdot s = x \cdot s'$ for all x implies $s = s'$. One can pass from the non-faithful to this case by identifying all $s, s' \in \Sigma$ having the same action on all $x \in X$.

It is easily checked that the wreath product operation is associative on the class of transition automata and that it preserves faithfulness.

For any set Z, let $P(Z)$ denote its power set. A *relational morphism* $R : (X, \Sigma) \lhd (Y, \Xi)$ is a pair of functions $\theta_R : X \to P(Y)$ and $\varphi_R : \Sigma \to P(\Xi)$ satisfying for all $x \in X$ and $s \in S$:

$$\theta(x) \neq \emptyset,\ \varphi(s) \neq \emptyset \tag{2.1}$$

$$y \in \theta(x), t \in \varphi(s) \Rightarrow y \cdot t \in \theta(x \cdot s) \tag{2.2}$$

One can naturally view θ as a subset of $X \times Y$ and φ as a subset of $\Sigma \times \Xi$. Indeed, one may consider $\theta = \{(x, y) \in X \times Y : y \in \theta(x)\}$ and similarly $\varphi = \{(s, t) \in \Sigma \times \Xi : t \in \varphi(s)\}$.[2] Notice how relational morphism is a generalization of *morphism* of transition automata, which is pair of functions $\theta : X \to Y$ and $\varphi : \Sigma \to \Xi$ such that $\theta(x) \cdot \varphi(s) = \theta(x \cdot s)$ always holds.

[2] This is source the term *relational*.

Relational morphisms are intimately related with the wreath product decomposition theory of automata.

We can compose relations in the usual way, yielding the composite of relational morphisms as the componentwise composite of their component relations. Namely, the composite RR' of relations R and R' is the smallest relation satisfying: xRx' and $x'R'x''$ implies $xRR'x''$. Also it becomes natural to write $x\theta y$ for $x \in \theta(y)$ (equivalently, $y \in \theta^{-1}(x)$, etc.), and to abbreviate condition (2.2) as $\theta \cdot \varphi \subseteq \theta$.

We call a relational morphism $R = (\theta, \varphi)$ *surjective* if $Y = \bigcup_{x\in X}\theta(x)$ and $\Xi = \bigcup_{s\in\Sigma}\varphi(s)$. Let $Im\ \theta$ denote $\bigcup_{x\in X}\theta(x)$, the image of θ. We call R *injective* if $\theta(x)\cap\theta(x') \neq \emptyset$ implies $x = x'$ and the analogous condition also holds for φ.

An injective relational morphism is also called an *emulation* or *covering* since it shows how to use (Y, Ξ) to emulate the computation of (X, Σ). Namely, lifting a state $x \in X$ to any $y \in \theta(x)$ and lifting transformation $s \in S$ to any $t \in \theta(s)$, we can compute $x \cdot s$ as follows. By definition of relational morphism $y \cdot t$ lies in $\theta(x \cdot s)$, but by injectivity in no other $\theta(x')$. Hence $x \cdot s$ is the unique element of X for which $y \cdot t$ is a lift. So $x \cdot s$ can be recovered after computing with any lifts in (Y, Ξ) of x and s. This means that (Y, Ξ) is computationally at least as powerful as (X, Σ). Any computation that can be done using (X, Σ) can be carried out using (Y, Ξ) via lifts given by the covering morphism R. We then say that (Y, Ξ) *covers* or (in computer science terminology) *emulates* (X, Σ).

For covering we use the notation, $(X, \Sigma) \prec (Y, \Xi)$, which is pronounced "ex-sigma divides why-xi "and call the covering relational morphism a *division.*

Note that a injective (resp. surjective) morphism is an injective (resp. surjective) relational morphism. A morphism is an *embedding* if its component functions are injective.

The reader should immediately verify the following easy facts:

Facts 1. (1) *A morphism of transition automata is a relational morphism.*
(2) *If R is a surjective relational morphism, then so is R^{-1}.*
(3) *The composite of relational morphisms is a relational morphism.*
(4) *The inverse of an embedding or surjective morphism is a covering.*

Fact 2. *If (X, Σ) divides $(Z, \Upsilon)\circ(Y, \Xi)$, then lifting to $(Z, \Upsilon)\circ(Y, \Xi)$ followed by projection to (Y, Ξ) is a relational morphism from (X, Σ) to (Y, Ξ).*

Proof: The division is a relational morphism. Projection is a morphism. Hence their composite is a relational morphism by facts 1.1 and 1.3 above. □

3. The Kernel of a Relational Morphism

The *kernel D_R of a relational morphism* $R : (X, \Sigma) \lhd (Y, \Xi)$ of transition automata is a structure with two types of objects, sets and arrows. D_R consists of a collection of sets $\theta^{-1}(y)$ as y ranges through the image of θ and collections of arrows $Arr_{y,t}$ of the form $[y, s, t]$ where $y \in Im\,\theta, t \in \varphi(s)$.

Observe that D_R is naturally a set with partial transformations, i.e. a partial automaton. The pairs of θ are states and elements of $Im\,\theta \times \varphi$ are the inputs. More precisely, D_R is the set of states (x, y), where $x \in \theta^{-1}(y)$, and input letters $[y', s, t]$, where $s \in \varphi^{-1}(t)$, determining partial transformations such that $(x, y) \cdot [y', s, t]$ is undefined for $y \neq y'$ but equals $(x \cdot s, y \cdot t)$ otherwise. Note the definition of relational morphism then guarantees that this $(x \cdot s, y \cdot t)$ is a state of D_R.

4. The Covering Lemma for Transition Automata

Let $R = (\theta, \varphi) : (X, \Sigma) \lhd (Y, \Xi)$ be a relational morphism of transition automata and D_R be its kernel as defined above. Let a transition automaton (Z, Υ) be given along with the following data:

(1) **(Lifting and Injectivity on States)** Injective relations $w_y : \theta^{-1}(y) \to P(Z)$ for each $y \in Im\,\theta$. (NB: For distinct $y, y' \in Y$ with $\theta^{-1}(y) = \theta^{-1}(y')$, we are allowed to have $w_y \neq w_{y'}$.)

(2) **(Lifting on Arrows)** For $y \in Im\,\theta$, $t \in Im\,\varphi$, relations $w_{y,t} : Arr_{y,t} \to P(\Upsilon)$ (where $Arr_{y,t}$ is the set of arrows $[y, s, t]$ of D_R) satisfying

$$\forall [y, s, t] \in Arr_{y,t},\ w_{y,t}([y, s, t]) \neq \emptyset$$

(Note: we may suppress the subscripts of $w_{y,t}$ when an argument is present.)

(3) **(Separation Property)** For each pair of distinct elements s, s' in Σ,

$$t \in \varphi(s) \cap \varphi(s') \Rightarrow \exists y \in Im\,\theta,\ w_{y,t}([y, s', t]) \cap w_{y,t}([y, s, t]) = \emptyset.$$

(4) **(Compatible Mapping)** For all $y \in Im\,\theta, t \in Im\,\varphi$, $s \in \varphi^{-1}(t)$,

$$w_y(x) \cdot w_{y,t}([y, s, t]) \subseteq w_{y \cdot t}(x \cdot s).$$

We then say that (Z, Υ) *computes* the kernel of R via the *labelling* w.

The schematic figure (Fig. 1) illustrates this notion. (Note that the sets shown as disjoint in the figure need not be.)

The condition

(3′) **(Injectivity on Arrows)** The relations $w_{y,t} : Arr_{y,t} \to P(\Upsilon)$ are injective.

implies condition (3):

Proposition 3. *Suppose $R : (X, \Sigma) \lhd (Y, \Xi)$ is a transition automaton, and let w be a labelling for D_R that satisfies the definition of "computes" except possibly the separation property. Then injectivity on arrows implies the separation property.*

Proof: Given $s \neq s'$ in Σ and $t \in \varphi(s) \cap \varphi(s')$. Let $y \in Im\,\theta$ be chosen arbitrarily. Notice that $s \neq s'$ implies $[y, s, t] \neq [y, s', t]$. From injectivity of $w_{y \cdot t}$, conclude that $w_{y,t}([y, s, t]) \cap w_{y,t}([y, s', t]) = \emptyset$ as required. □

Theorem 4 (Covering Lemma for Transition Automata).

I. *Let R be a relational morphism $(X, \Sigma) \lhd (Y, \Xi)$ and let (Z, Υ) compute D_R via a labelling w. Then we construct a covering*

$$(X, \Sigma) \prec (Z, \Upsilon) \circ (Y, \Xi).$$

Moreover, R is the composite of the covering and projection to (Y, Ξ).

II. *Let (X, Σ) be faithful. If (X, Σ) divides $(Z, \Upsilon) \circ (Y, \Xi)$, then the relational morphism R obtained by composing the division with the projection onto (Y, Ξ) has kernel D_R computed by (Z, Υ).*

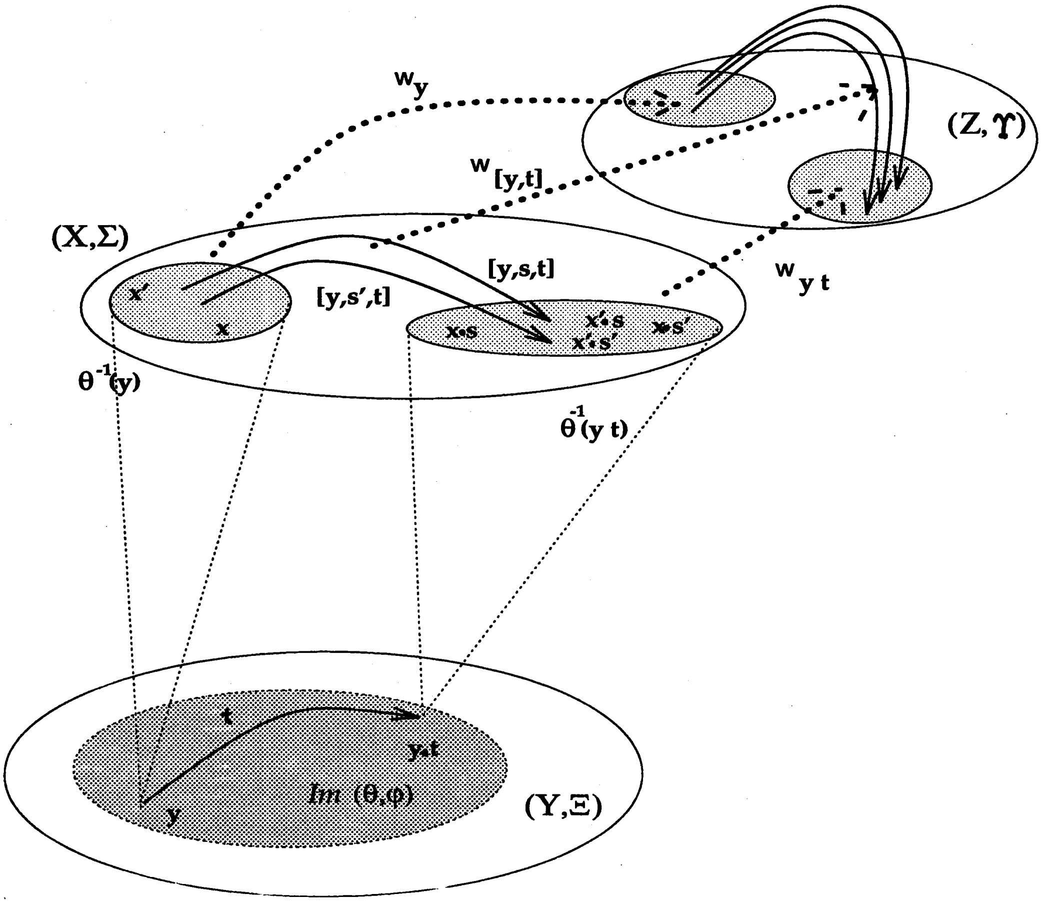

FIGURE 1. A Labelling w of the Kernel $D_{\theta,\varphi}$ computed by (Z,Υ)

Proof of I: Let $R = (\theta,\varphi) : (X,\Sigma) \lhd (Y,\Xi)$ be a relational morphism whose kernel D_R is computed by w. We define a covering morphism $(\psi,\mu) : (X,\Sigma) \prec (Z,\Upsilon) \circ (Y,\Xi)$ as follows:

$$\psi(x) = \{(z,y) \in Z \times Y : z \in w_y(x), y \in \theta(x)\} \tag{4.3}$$

$$\mu(s) = \{(f,t) \in \Upsilon^Y \times \Xi : t \in \varphi(s),\ \forall y \in Im\,\theta,\ f(y) \in w_{y,t}([y,s,t])\} \tag{4.4}$$

Obviously $\psi(x) \neq \emptyset$ and $\mu(s) \neq \emptyset$.

In (4.4) note that for $y \in Y \setminus Im\,\theta$, the value $f(y)$ may be taken to be any $u \in \Upsilon$.

Then for (z,y) in $\psi(x)$ and (f,t) in $\mu(s)$, we have:

$$(z,y) \cdot (f,t) = (z \cdot f(y), y \cdot t).$$

Since R is a relational morphism

$$y \cdot t \in \theta(x \cdot s),$$

and using the definition of 'computes' (mapping compatibility)

$$z \cdot f(y) \in w_y(x) \cdot w_{y,t}([y,s,t]) \subseteq w_{y \cdot t}(x \cdot s).$$

Whence $(z,y) \cdot (f,t)$ lies in $\psi(x \cdot s)$, showing

$$\psi(x) \cdot \mu(s) \subseteq \psi(x \cdot s),$$

as required. Thus $(\psi, \mu) : (X, \Sigma) \lhd (Z, \Upsilon) \circ (Y, \Xi)$ satisfies the definition of a relational morphism.

We must still verify (ψ, μ) is an injective relational morphism: *For states*, suppose $(z,y) \in \psi(x) \cap \psi(x')$, then $z \in w_y(x) \cap w_y(x')$. But w_y is an injective relation, so $x = x'$. Hence, ψ is injective. *For transformations*, suppose $(f,t) \in \mu(s) \cap \mu(s')$. Then $t \in \varphi(s) \cap \varphi(s')$. If $s \neq s'$, there exists a $y \in Im\,\theta$ as in the separation property, *i.e.* $w([y,s,t]) \cap w([y,s',t]) = \emptyset$. But by definition of μ, $f(y)$ lies in this intersection. Hence it must be that $s = s'$. Thus μ is an injective relation, so our relational morphism is a covering.

Let $p : (Z, \Upsilon) \circ (Y, \Xi) \twoheadrightarrow (Y, \Xi)$ be the projection: $p(z,y) = y$ and $p(f,t) = t$. Obviously $p(\psi(x)) = \theta(x)$ and $\varphi(s) = p(\mu(s))$.

Proof of II: Conversely, suppose we have a covering $(\psi, \mu) : (X, \Sigma) \prec (Z, \Upsilon) \circ (Y, \Xi)$. Let p be the projection to (Y, Ξ), and let $R = (\theta = p \circ \psi, \varphi = p \circ \mu)$ be the induced relational morphism $(X, \Sigma) \lhd (Y, \Xi)$. We construct a labelling w which shows (Z, Υ) computes the kernel D_R of R:

Lifting and Injectivity on States: If $y \in Im\,\theta$, we define for $x \in \theta^{-1}(y)$,

$$w_y(x) = \{z \in Z : (z,y) \in \psi(x)\},$$

which is always non-empty.

Since ψ is an injective relation, $(z,y) \in \psi(x) \cap \psi(x')$ implies $x = x'$, hence $w_y : \theta^{-1}(y) \longrightarrow P(Z)$ is an injective relation.

Lifting on Arrows: For an arrow $[y,s,t]$ in D_R, we have $t \in p(\mu(s)) = \varphi(s)$, and define

$$w_{y,t}([y,s,t]) = \{f(y) \in \Upsilon : (f,t) \in \mu(s)\}.$$

Notice this is well-defined and non-empty.

Separation property: Assume $t \in \varphi(s_1) \cap \varphi(s_2)$. And suppose for all $y \in Im\,\theta$, there is a $g(y) \in w_{y,t}([y,s_1,t]) \cap w_{y,t}([y,s_2,t])$. Now suppose (f_i,t) are lifts of s_i for $i = 1,2$. Take any lift (z,y) of any $x \in X$. We claim $x \cdot s_1 = x \cdot s_2$: By definition of w, $w([y,s_i,t])$ contains $g(y)$ for $i = 1,2$. So for $i = 1,2$ there exist $(f_i,t) \in \mu(s_i)$ with $f_i(y) = g(y)$. Therefore $(z,y)(f_i,t) = (z \cdot g(y), y \cdot t)$ is a lift of $x \cdot s_i$. Hence $(z \cdot g(y), y \cdot t)$ lies in both $\psi(x \cdot s_1)$ and $\psi(x \cdot s_2)$. Therefore $x \cdot s_1 = x \cdot s_2$ by injectivity of ψ. Since x was chosen as an arbitrary $x \in X$, we conclude from faithfulness of (X, Σ) that $s_1 = s_2$.

Finally we check *mapping compatibility:*

$$\begin{aligned} w_y(x) \cdot w_{y,t}([y,s,t]) &= \{z \cdot f(y) : (z,y) \in \psi(x);\ (f,t) \in \mu(s)\} \\ &= \{z' : (z', y \cdot t) \in \psi(x) \cdot \mu(s)\} \\ &\subseteq \{z' : (z', y \cdot t) \in \psi(x \cdot s)\} \\ &= w_{y \cdot t}(x \cdot s). \end{aligned} \qquad \square$$

5. Embedding Computations of Transition Automata

Let $R = (\theta, \varphi)$ be a surjective morphism. We call a labelling w computing the kernel D_R via some transition automaton (Z, Υ) an *embedding computation* if the sets $w_y(x)$ and $w_{y,t}([y, s, t])$ are singletons whenever defined.

Theorem 5 (Embedding Computation Theorem for Transition Automata). *Let a surjective morphism* $R : (X, \Sigma) \overset{\theta,\varphi}{\twoheadrightarrow} (Y, \Xi)$ *be given. A labelling* w *of* D_R *is an embedding computation by* (Z, Υ) *if and only if*

$$(X, \Sigma) \textit{ embeds into } (Z, \Upsilon) \circ (Y, \Xi)$$

and the projection to (Y, Ξ) *induces* R.

Proof: Given the embedding computation labelling for D_R, the proof of the Covering Lemma yields a covering

$$(\psi, \mu) : (X, \Sigma) \prec (Z, \Upsilon) \circ (Y, \Xi)$$

as described above, with projection inducing R. Now for all $x \in X$, $\psi(x) = \{(z, y) : z \in w_y(x), y = \theta(x)\}$ is a singleton, as is $\mu(s) = \{(f, t) : f(y) \in w_{y,t}([y, s, t]), t = \varphi(s)\}$ for each $s \in \Sigma$ (since θ and φ are functions and θ is onto). And since this is a covering, $\psi(x) \cdot \mu(s)$ is contained in – hence coincides with – the singleton $\psi(x \cdot s)$.

By injectivity of ψ as a relation, it now follows ψ is injective when viewed as a function from X to $Z \times Y$. Similarly μ is injective as a function. Thus (ψ, μ) is an embedding of transition automata.

Conversely, given an embedding inducing R, the w as constructed in the proof for part II of the Covering Lemma computes D_R: Whether or not (X, Σ) is faithful, the separation property in this case is easily checked (use injectivity of μ and surjectivity of θ). The proof that the other properties in the definition of 'computes' hold does not require faithfulness and is exactly as in the theorem's proof. Since the $w_{y,t}([y, s, t])$ and $w_y(x)$ are always singletons, this is an embedding computation. □

Remarks: (1) In applying the Embedding Computation Theorem, one may like to obtain a surjective morphism from a given morphism by restricting the target to be the morphism's image. (2) Also if embedding followed by projection as in the theorem's converse is not surjective, then (Y, Ξ) may be replaced by the image (Y', Ξ') of (X, Σ) and we still have an embedding of (X, Σ) into $(Z, \Upsilon) \circ (Y', \Xi')$ by restricting the action of images of elements of Σ to $Z \times Y'$. However, (Y', Ξ') may fail to be faithful even when (Y, Ξ) is.

References

1. B. Austin, K. Henckell, C. Nehaniv, and J. Rhodes. Subsemigroups and Complexity via the Presentation Lemma. *Journal of Pure and Applied Algebra*, 1995, (*in press*).
2. C. L. Nehaniv. From relation to emulation: The covering lemma for transformation semigroups. In Y. Kobayashi, editor, *Proceedings of 18th Annual Meeting on Semigroups, Formal Languages and Their Related Areas, Tokyo, Japan*, Also to appear in *Journal of Pure and Applied Algebra.*
3. John Rhodes and Bret Tilson. The kernel of monoid morphisms. *Journal of Pure and Applied Algebra*, 62:227–268, 1989.
4. B. Tilson. Categories as algebras: an essential ingredient in the theory of monoids. *Journal of Pure and Applied Algebra*, 48:83–198, 1987.