Flocks, groups and heaps, joined with semilattices

Arthur Knoebel

Abstract. This article describes the lattice of varieties generated by those of flocks and near heaps. Flocks and heaps are two ways of presenting groups by a ternary operation rather than a binary one. Their varieties joined with that of ternary semilattices create the varieties of near flocks and near heaps. This is done by finding normal forms for words that make up free algebras. Simple sets of identities define these varieties. Identities in general are decidable. Each near flock is a Płonka sum of flocks, and each near heap is a Płonka sum of heaps. An algorithm translates any binary group identity to one in a ternary operation satisfied by near heaps.

1. Introduction

This article merges groups, which arise from composing permutations, with semilattices, which are partial orders with least upper bounds. This is not done by imposing an order on groups, of which there is an extensive literature, but by joining their varieties. The varietal join of groups with semilattices is achieved seamlessly with operations having three arguments instead of the usual two. There are several ways to do this. We look at flocks and heaps.

Figure 1 shows the lattice of these varieties. In each box are the algebras, the name of their variety, and the identities defining it. The top box, containing all of the varieties, is the new variety of near flocks. We prove that these varieties are related as depicted, and decompose algebras higher up into those lower down, wherever possible.

The traditional binary operation $\times$ of a group has five desirable properties: associativity, unique solvability in each argument, and hence the existence of a unity, from which follow inverses and cancellation. There are several ways to change the binary operation to a ternary one $[,,]$, where these properties diverge. Our way defines it by:

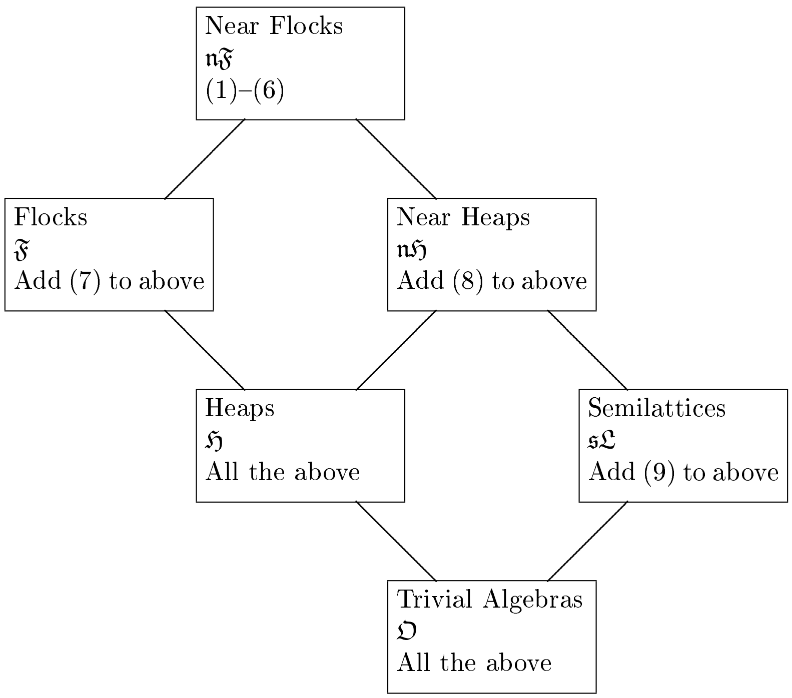
$$[x, y, z] = (x \times y^{-1}) \times z.$$

This satisfies (1) and (2), which together are called the *para-associative law.* This operation is also uniquely solvable in each argument, which implies cancellation. However there is neither a unity nor an inverse operation.

A set with a ternary operation satisfying the para-associative law and being solvable in each argument is a flock in the original sense. But solvability is not

2010 Mathematics Subject Classification: 06A12, 08A05, 08B05, 08B20, 20N10, 20N15
Keywords: near flock, near heap, group, semilattice, join, variety, identity, extension, Płonka sum.

definable by identities. To do so requires adding a unary operation $\overline{x}$ that captures regularity in the sense of von Neumann: for each x, the element $\overline{x}$ satisfies $[x, \overline{x}, x] = x$. This is not necessarily the inverse, although it may be is some cases, and in other cases it may be the identity function. When $\overline{x}$ is the identity function we have heaps.



$$[v, w, [x, y, z]] \approx [[v, w, x], y, z], \tag{1}$$
$$[v, w, [x, y, z]] \approx [v, [y, x, w], z], \tag{2}$$
$$\overline{[x, y, z]} \approx [\overline{x}, \overline{y}, \overline{z}], \tag{3}$$
$$\overline{\overline{x}} \approx x, \tag{4}$$
$$[x, \overline{x}, x] \approx x, \tag{5}$$
$$[x, \overline{x}, y] \approx [\overline{x}, x, y] \approx [y, x, \overline{x}] \approx [y, \overline{x}, x], \tag{6}$$
$$[x, \overline{x}, y] \approx y, \tag{7}$$
$$\overline{x} \approx x, \tag{8}$$
$$[x, x, y] \approx [x, y, y]. \tag{9}$$

Figure 1. Lattice of varieties of flocks, heaps and semilattices, and their defining identities.

The binary operation $\wedge$ of semilattices may be turned into a ternary operation:

$$[x, y, z] = (x \wedge y) \wedge z.$$

Then the equation $[x, y, x] = x$ always has the solution $y = x$ although it may not be unique. Nevertheless we set $\overline{x} = x$, the identity function. Although it could be dropped in semilattices and heaps, for uniformity in comparing varieties we type algebras with a ternary operation and a unary operation, that is the type is $< 3, 1 >$, except where otherwise noted later in this article.

For groups there is also the triple composition:

$$[x, y, z] = (x \times y) \times z,$$

without a middle inverse y^{-1}. This satisfies associative laws and solvability. The inverse operation $\overline{x} = x^{-1}$ may be again added to the type as the solution to $[x, \overline{x}, x] = x$. But this is not the main line of investigation, and will be passed over.

The lack of a unity is no loss and may be an advantage. In the study of vector spaces, where a base-free presentation favors no particular axes, just as in the physical world no particular directions are preferred, so a presentation with no origin should be applauded, as it goes along with the universe having no designated center. Still, the ternary operation has a physical meaning, at least for vector spaces, it is the completion of a parallelogram, that is, it is the fourth vertex, $d = [a, b, c]$, of a parallelogram when the other three vertices are a, b and c.

With the definition of the varieties by ever increasing sets of identities as we go downward in Figure 1, it is clear that the lines represent set-theoretical inclusion as we go upwards. It remains to be proven that the joins and meets are varietal: for example, that $\mathfrak{nF}$ is the smallest variety that includes both $\mathfrak{F}$ and $\mathfrak{nH}$, and that $\mathfrak{H}$ is the largest variety included in both $\mathfrak{F}$ and $\mathfrak{nH}$.

To do this for joins, we find for each variety a normal form for its terms. These constitute the free algebras. The identities in each variety are decidable.

Algebras in the joins are built from algebras below them. A near flock is a Płonka sum of flocks, which is a special kind of extension of flocks by a semilattice. A near heap is a Płonka sum of heaps.

Through the next four sections we descend from the top of the lattice of Figure 1. Since the operation $\overline{x}$ has no effect on the variety $\mathfrak{nH}$, because of identity (8), it will eventually be left out in the treatment of the varieties lower in the lattice.

The next to last section spells out the close connection between heaps and groups as an adjoint situation that is almost a categorical equivalence. The last section translates any group identity to its counterpart in heaps.

2. Near flocks

The variety $\mathfrak{nF}$ of near flocks is defined by the set nF of identities (1)–(6), and is at the top of the lattice of Figure 1. Free algebras are built with normal words.

With these it will be proven in the next section that the variety of near flocks is the join of those of flocks and near heaps.

First, we derive some consequences of the identities defining near flocks. Only some of what is needed is written out here. More identities may be manufactured by their reflection. The *reflection* of an identity is it written backwards, literally. For instance, the reflection of (2), $[vw[xyz]] \approx [v[yxw]z]$, is $[z[wxy]v] \approx [[zyx]wv]$. Since the reflections of (1)–(6) are consequences of these axioms, a reflection of any consequence of (1)–(6) is also a consequence of them.

Proposition 2.1. *These identities for near flocks follow from* $(1)-(6)$.

$$[[w,x,\overline{x}],y,z] \approx [[w,y,z],x,\overline{x}] \tag{10}$$

$$[[w,y,x],\overline{x},z] \approx [[w,y,z],x,\overline{x}], \tag{11}$$

$$[[x,y,z],\overline{[x,y,z]},w] \approx [x,\overline{x},[y,\overline{y},[z,\overline{z},w]]]. \tag{12}$$

$$[[x,v,w],[y,\overline{v},\overline{w}],[z,v,w]] \approx [[x,y,z],v,w]. \tag{13}$$

Proof.

(10).
$$\begin{aligned}
[[w,x,\overline{x}],y,z] &\approx [[x,\overline{x},w],y,z] && (6)\\
&\approx [x,\overline{x},[w,y,z]] && (1)\\
&\approx [[w,y,z],x,\overline{x}] && (6).
\end{aligned}$$

(11).
$$\begin{aligned}
[[w,y,x],\overline{x},z] &\approx [w,y,[x,\overline{x},z]] && (1)\\
&\approx [w,y,[z,x,\overline{x}]] && (6)\\
&\approx [[w,y,z],x,\overline{x}] && (1).
\end{aligned}$$

(12).
$$\begin{aligned}
[[x,y,z],\overline{[x,y,z]},w] &\approx [[x,y,z],[\overline{x},\overline{y},\overline{z}],w] && (3)\\
&\approx [[[x,y,z],\overline{z},\overline{y}],\overline{x},w] && (2)\\
&\approx [[[x,y,\overline{y}],\overline{x},w],z,\overline{z}] && (11),(10)\\
&\approx [[x,y,\overline{y}],\overline{x},[z,\overline{z},w]] && (1),(6)\\
&\approx [x,\overline{x},[y,\overline{y},[z,\overline{z},w]]] && (1),(11).
\end{aligned}$$

(13).
$$\begin{aligned}
[[x,v,w],[y,\overline{v},\overline{w}],[z,v,w]] &\approx [[[x,v,w],\overline{w},\overline{v}],y,[z,v,w]] && (2)\\
&\approx [[[[x,v,w],\overline{w},\overline{v}],y,z],v,w] && (1)\\
&\approx [[[[x,y,z],v,\overline{v}],v,w],\overline{w},w] && (10),(11)\\
&\approx [[x,y,z],v,w] && (1),(5).\ \square
\end{aligned}$$

Identities (10) and (11) of this proposition suggest isolating pairs of adjacent variables when one is barred and the other is not.

Normal near flock words, introduced in the next definition, will serve as the elements of free near flocks. A distinction is made between terms and words.

Definition 2.2. In contrast to a term, built with variables and operations symbols, a *word* is simply a finite string of these letters with no ternary operation symbols but with single bars over some of the letters. For example, $\left[x_2, \overline{\overline{x}}_1, \overline{[x_4, x_1, \overline{x}_4]}\right]$ is a term, and $x_2 x_1 \overline{x}_4 \overline{x}_1 x_4$ is a word. A letter adjacent to itself barred, $x_i \overline{x}_i$ or $\overline{x}_i x_i$, is called a *skew pair*. Let $|w|$ be the *length* of a word, that is, the number of occurrences of letters in it. A *normal near flock word*, or simply normal word in this section, is a word, $w = w^\phi w^\sigma$, with two parts, namely a flock part w^ϕ and a semilattice part w^σ — their names will be motivated later. The *flock part* w^ϕ is of odd length in which no variable x_i and its bar $\overline{x}_i$ are adjacent, in either order. The *semilattice* or *skew part* w^σ is of even length and is a sequence, $x_{i_1}\overline{x}_{i_1}x_{i_2}\overline{x}_{i_2}\dots x_{i_k}\overline{x}_{i_k}$, of skew pairs $x_i\overline{x}_i$ with the indices in increasing order: $i_1 < i_2 < \dots < i_k$. All letters occurring in the flock part w^ϕ must occur in the semilattice part w^σ, but not all the letters in w^σ need be in w^ϕ. For example, here are the parts of a normal word:

$$
\begin{aligned}
w &= x_5 x_5 x_2 \overline{x}_5 x_6 \overline{x}_2 \overline{x}_2 \; x_2 \overline{x}_2 x_5 \overline{x}_5 x_6 \overline{x}_6 x_9 \overline{x}_9, \\
w^\phi &= x_5 x_5 x_2 \overline{x}_5 x_6 \overline{x}_2 \overline{x}_2, \\
w^\sigma &= x_2 \overline{x}_2 x_5 \overline{x}_5 x_6 \overline{x}_6 x_9 \overline{x}_9.
\end{aligned}
$$

Definition 2.3. To define free near flocks we need to manipulate words with some operators: the first operator w^ρ reverses the order of the variables; for example, $(x_1 x_2 \overline{x}_3)^\rho$ is $\overline{x}_3 x_2 x_1$. Note that $(uvw)^\rho = w^\rho v^\rho u^\rho$ for words u, v and w. The second operator $\Cup$ joins semilattice parts: $v^\sigma \Cup w^\sigma$ is the string of all skew pairs $x_i\overline{x_i}$ for all variables x_i in v or w, put in order of increasing index with no skew pair occurring more than once. The third transforms a word w of odd length back into a term w^β by appropriately inserting pairs of brackets to form ternary operations all associated to the left; for example, the word $w = x_5 x_5 x_2 x_2 \overline{x}_5 x_6 \overline{x}_2$ becomes the term $w^\beta = [[[x_5, x_5, x_2], x_2, \overline{x}_5], x_6, \overline{x}_2]$. The fourth is an algorithm, given in the next definition, that normalizes any term.

Definition 2.4. Here is how to turn any near flock term t into a normal near flock word t^ν by using (1)–(6). Let $x_{i_1}, x_{i_2}, \dots, x_{i_n}$ be the variables of t with $i_1 < i_2 < \dots < i_n$. As a running example, consider $t = [[x_2, \overline{\overline{x_2}}, x_1], \overline{[\overline{x}_1, x_2, x_1]}, \overline{x}_1]$. Use (3) to push all bars of t onto individual variables, and (4) to eliminate more than one bar on a variable. The example becomes $[[x_2, x_2, x_1], [x_1, \overline{x}_2, \overline{x}_1], \overline{x}_1]$.

With (5) create a skew pair $x_i\overline{x}_i$ for each variable x_i in t not already in such a pair. Use (10) and (11) to move these skew pairs, one at a time, to the extreme right, in order of increasing index. The example has now become

$$[[[[x_2, x_2, x_1], [x_1, \overline{x}_2, \overline{x}_1], \overline{x}_1], \; x_1, \overline{x}_1], x_2, \overline{x}_2].$$

No skew pair will now be across a bracket. Then use (1) and (2) to associate all occurrences of $[,,]$ to the far left. So we have

$$[[[[[x_2, x_2, x_1], \overline{x}_1, \overline{x}_2], x_1, \overline{x}_1], \; x_1, \overline{x}_1], x_2, \overline{x}_2].$$

At this point we may as well remove the brackets and work with the resulting word $w = w^\upsilon w^\sigma$. Here w^σ is the semilattice part — the word $x_{i_1}\overline{x}_{i_1}x_{i_2}\overline{x}_{i_2}\ldots x_{i_n}\overline{x}_{i_n}$ of all skew pairs that have been moved. The remainder w^υ of w on the right will be reworked to give the flock part. The example turns into a word with parts

$$w^\upsilon = x_2x_2x_1\overline{x}_1\overline{x}_2x_1\overline{x}_1 \text{ and } w^\sigma = x_1\overline{x}_1x_2\overline{x}_2.$$

Remember that at any time the operator $^\beta$ can return brackets, associated to the left.

Up to now the algorithm has been deterministic; there have been no choices that might make a difference. But new skew pairs may appear in w^υ as a result of having moved old ones to the right; for example, when $x_1\overline{x}_1$ is removed from the middle of w^υ, then $x_2\overline{x}_2$ is a new skew pair. Now there will be choices as to which order to eliminate these unnecessary skew pairs. The next proposition will show that these choices make no difference in the final outcome.

With (10) and (11) move skew pairs as they appear over to their corresponding pairs in w^σ on the right, and eliminate duplicates with (5). The remaining part of the word on the left, with no skew pairs, is the desired flock part w^ϕ. Our term, $t = [[x_2, \overline{\overline{x_2}}, x_1], \overline{[\overline{x}_1, x_2, x_1]}, \overline{x}_1]$ has become the normal word $t^\nu = x_2\ x_1\overline{x}_1x_2\overline{x}_2$.

Proposition 2.5. *The outcome of the algorithm of Definition* 2.4 *does not depend on the order of eliminating skew pairs.*

Proof. By induction on the length $|w^\upsilon|$ of what is to become the flock part of a word w. Suppose the proposition is true when the length is less than n. Assume a particular w^υ has length n. Consider sequences, $p = \langle p_1, p_2, \ldots\rangle$, of occurrences p_i of skew pairs that appear in it and that are being successively eliminated; call them *paths*. Think of two different paths, $p = \langle p_1, p_2, \ldots\rangle$ and $q = \langle q_1, q_2, \ldots\rangle$. We will show that, after removing the skew pairs in them, we arrive at the same flock part w^ϕ. There are three possibilities for the first pairs: p_1 and q_1 are the same; p_1 and q_1 are not the same but overlap; p_1 and q_1 do not overlap, that is, they are disjoint. We dispose of these possibilities in order.

Suppose that p_1 is q_1, and this skew pair is eliminated from both paths, Then the remaining words will be the same and have length less than n. By the induction hypothesis, after all the remaining skew pairs are eliminated from the two paths, we end up with the same word.

Next suppose that p_1 and q_1 overlap, that is, we have for example $x_i\overline{x}_ix_i$ in w^υ with p_1 being $x_i\overline{x}_i$ and q_1 being $\overline{x}_ix_i$. Eliminating either skew pair leaves the same word of lesser length, and the induction hypothesis applies again.

Now assume the two paths start out with disjoint skew pairs, that is, a path can start out at either p_1 or q_1, which are not the same. In particular new paths can start as $p' = \langle p_1, q_1, \ldots\rangle$ and $q' = \langle q_1, p_1, \ldots\rangle$. Now, by the induction hypothesis, the elimination of the same first skew pairs of p and p' will end up with the same word, since they start out the same; and so will q and q'. As p_1 and q_1 are disjoint, what is left after they are both removed is the same word r. Its length $|r|$ is less than n, and hence we are led to the same normal word no matter in what order skew pairs of r are thrown out. So p' and q' will terminate the algorithm at the same flock part, and hence so will p and q. $\square$

We now extend the use of $^{\phi}$ from designating the flock part of a normal word to its use as an operator that creates the normal flock part from any word of odd length. Similarly $^{\sigma}$ becomes the operator that creates the semilattice part

Definition 2.6. Write $F_{\mathfrak{n}\mathfrak{F}}(\alpha)$ for the set of all normal words for near flocks, each with a finite number of letters from the set $\{x_i \mid i < \alpha\}$. Turn this into an algebra of type $\langle 3, 1\rangle$, soon to be proven a near flock. For normal words u, v, w define the flock part $[u, v, w]^{\phi}$ of the operation $[u, v, w]$ to be the string $(u^{\phi} v^{\phi\rho} w^{\phi})^{\phi}$, and the semilattice part $[u, v, w]^{\sigma}$ to be $u^{\sigma} \Cup v^{\sigma} \Cup w^{\sigma}$. (The operation $\Cup$ is associative; see Definition 2.3.) The operation $\overline{w}$ adds bars to those variables in the flock part that have none and removes bars from those that do, it leaves the semilattice part alone. Write $\boldsymbol{F}_{\mathfrak{n}\mathfrak{F}}(\alpha)$ for the algebra $\langle F_{\mathfrak{n}\mathfrak{F}}(\alpha); [\,,\,], ^{-}\rangle$.

Proposition 2.7. *For each nonzero cardinal α, $\boldsymbol{F}_{\mathfrak{n}\mathfrak{F}}(\alpha)$ is a near flock.*

Proof. We prove that the identities (2), (3) and (4) are satisfied in $\boldsymbol{F}_{\mathfrak{n}\mathfrak{F}}(\alpha)$; the others are proven similarly.

(2). Let u, v, w, x, y and z be normal words. It suffices to prove (2) separately on the flock parts and the semilattice parts of words.

For the flock parts on each side of (2), we expand them to a common word:

$$[v, w, [x, y, z]]^{\phi} = (v^{\phi} w^{\phi\rho} [x, y, z]^{\phi})^{\phi} = (v^{\phi} w^{\phi\rho} (x^{\phi} y^{\phi\rho} z^{\phi})^{\phi})^{\phi} = (v^{\phi} w^{\phi\rho} x^{\phi} y^{\phi\rho} z^{\phi})^{\phi},$$

$$[v, [y, x, w], z]^{\phi} = (v^{\phi} [yxw]^{\phi\rho} z^{\phi})^{\phi} = (v^{\phi} (y^{\phi} x^{\phi\rho} w^{\phi})^{\phi\rho} z^{\phi})^{\phi} = (v^{\phi} w^{\phi\rho} x^{\phi} y^{\phi\rho} z^{\phi})^{\phi}.$$

In the first line, Proposition **??** tells us that $(uvw^{\phi})^{\phi} = (uvw)^{\phi}$ for words u, v, w. In the second line, we also use the fact that $w^{\phi\rho} = w^{\rho\phi}$.

The semilattice parts are equal since $\Cup$ is associative and commutative.

(3) $\overline{[u, v, w]} = [\overline{u}, \overline{v}, \overline{w}]$. For the flock part this follows from the fact that $\overline{w}^{\phi} = \overline{w^{\phi}}$. So each path eliminating skew pairs from w has a corresponding path in $\overline{w}$. For the semilattice parts of u, v, w, the bar has no effect.

(4) $\overline{\overline{w}} = w$. The operation $^{-}$ toggles the bar operation on the flock part, leaving the semilattice part alone. $\square$

Theorem 2.8. *For each nonzero cardinal α, $\boldsymbol{F}_{\mathfrak{n}\mathfrak{F}}(\alpha)$ is the free near flock on α generators.*

Proof. We verify the universal property that characterizes free algebras: for any near flock $\boldsymbol{A}$ generated by α elements a_i $(i < \alpha)$, there is a unique homomorphism h from $\boldsymbol{F}_{\mathfrak{n}\mathfrak{F}}(\alpha)$ to $\boldsymbol{A}$ such that $h(x_i) = a_i$. To that end define $h : F_{\mathfrak{n}\mathfrak{F}}(\alpha) \to A$ by $h(w) = w^{\beta}(a_{i_1}, \ldots, a_{i_n})$ where $x_{i_1}, \ldots, x_{i_n}$ are the letters of the normal word w.

First we prove that h is a homomorphism, that is, it preserves the operations; then we prove that it is unique. Preserving the operation $[,,]$ means that

$$h([u,v,w]) = [h(u), h(v), h(w)] \qquad (u,v,w \in F_{\mathfrak{nF}}(\alpha)),$$

which is equivalent to

$$[u,v,w]^\beta \approx [u^\beta, v^\beta, w^\beta]. \tag{14}$$

This last requires a proof by induction on the length $|v|$ of the middle argument v.

If v is a single variable y, then identity (1) only, when applied to the left side of (14), will move all brackets of the normal word w to the left without any need of reversals by (2). Now suppose $v = [x,y,z]$ with x,y,z normal near flock words of length less than v. We calculate that

$$\begin{aligned}
[u,v,w]^\beta &= [u,[x,y,z],w]^\beta \\
&= [[u,z,y],x,w]^\beta && (2) \\
&= [[u^\beta,z^\beta,y^\beta],x^\beta,w^\beta] && \text{(induction hypothesis twice)} \\
&= [u^\beta,[x^\beta,y^\beta,z^\beta],w^\beta] && (2) \\
&= [[u^\beta,[x,y,z]^\beta,w^\beta] && \text{(induction hypothesis)} \\
&= [u^\beta,v^\beta,w^\beta].
\end{aligned}$$

Since the bar $^-$ does not change the order of the variables, h preserves it:

$$h(\overline{w}) = \overline{w}^\beta = \overline{w^\beta} = \overline{h(w)}.$$

To show ϕ is unique, let $g : F_{\mathfrak{nF}}(\alpha) \to A$ be another homomorphism such that $g(x_i) = a_i$ $(i < \alpha)$. Then, if $x_{i_1}, \ldots, x_{i_n}$ are the letters of a normal word w, we have that $g(w) = w^\beta(a_{i_1}, \ldots, a_{i_n}) = h(w)$, since a homomorphism preserves terms. □

Write nF for the set of identities (1) – (6) defining near flocks.

Proposition 2.9.

(*a*) *For any near flock term t there exists a unique normal near flock word w such that $\mathsf{nF} \vdash t \approx w^\beta$.*

(*b*) *For any normal near flock words v and w, $\mathsf{nF} \vdash v^\beta \approx w^\beta$ iff $v = w$.*

Proof. Existence falls out of Definition 2.4. Uniqueness follows from Theorem 2.8: as normal words, like v and w, make up $\boldsymbol{F}_{\mathfrak{nF}}(\alpha)$, a free near flock of the variety defined by the identities of nF, we conclude (*b*), from which follows uniqueness. □

Corollary 2.10. *The equational theory of near flocks is decidable.*

Proof. From (*b*) of Proposition 2.9, for terms t_1 and t_2 of type $\langle 3,1\rangle$,

$$\mathsf{nF} \vdash t_1 \approx t_2 \text{ iff } t_1^\nu = t_2^\nu.$$

This is true since $\mathsf{nF} \vdash t^{\nu\beta} \approx t$. Here t^ν is the normal near flock word obtained from a term t by the algorithm of Definition 2.4. □

3. Flocks

The set F of identities (1)–(7) define the variety $\mathfrak{F}$ of flocks. Originally flocks were defined by Dudek [6] as a nonempty set A with a ternary operation $[,,]$ that satisfies (1)–(2) and is uniquely solvable in each argument: for all a, b and c in A there are unique x, y, z such that $[x,a,b]=c$, $[a,y,b]=c$, and $[a,b,z]=c$. It is not possible to define unique solvability by identities in $[,,]$ alone without additional operations (see the end of this section for why not). However, by [6, Proposition 3.2], unique solvability does allow us to define a unary operation $^-$:

$$\overline{x} \text{ is the unique } y \text{ such that } [x,y,x]=x.$$

Adding (7) to those identities defining near flocks simplifies the theory since (7) allows all skew pairs to be removed from normal flock words. With them free algebras are defined and used to prove that the variety of near flocks is the join of flocks and near heaps. Finally, it is shown that each near flock is a Płonka sum of flocks by a semilattice.

Definition 3.1. As before, *words* are strings of letters, some with bars. But now, a *normal word* for flocks is a word of odd length in which no skew pairs occur, that is, neither $x_i\overline{x}_i$ nor $\overline{x}_ix_i$ occur. They are merely the flock parts of normal near flock words.

We pass between terms and words similarly to what was done in the last section. A normal flock word is obtained from any term by using the identities (1)–(6) to push all bars onto individual letters and eliminate multiple bars. Identities (10), (11) and (7) eliminate skew pairs. Identities (1) and (2) associate brackets to the left. With brackets removed, this is a normal flock word.

Definition 3.2. To define the *free flock* on α generators, let $F_{\mathfrak{F}}(\alpha)$ be the set of normal flock words on the set of α letters $\{x_i \mid i<\alpha\}$. Then $\boldsymbol{F}_{\mathfrak{F}}(\alpha)$ is the algebra $\langle F_{\mathfrak{F}}(\alpha);\ [,,],\ ^-\rangle$ of type $\langle 3,1\rangle$. Here, for normal flock words, u, v and w, the ternary operation $[u,v,w]$ is the catenation of them, $(uv^{\rho}w)^{\phi}$, with the order of the letters in the middle argument v reversed to v^{ρ}. This is followed by erasing any skew pairs that arise. The unary operation $\overline{w}$ removes bars from letters in w that have them and adds them otherwise.

The next proposition is on the way to showing that $\boldsymbol{F}_{\mathfrak{F}}(\alpha)$ is a free flock.

Proposition 3.3. *For any non-zero cardinal α, $\boldsymbol{F}_{\mathfrak{F}}(\alpha)$ is a flock.*

Proof. Axiom (7) is satisfied since, for normal flock words w and $v=x_{i_1}\ldots x_{i_n}$,

$$[v,\overline{v},w]=(v\overline{v}^{\rho}w)^{\phi}=(x_{i_1}\ldots x_{i_n}\overline{x}_{i_n}\ldots\overline{x}_{i_1}w)^{\phi}=w^{\phi}=w.$$

Cancelling inner letters by the operator ϕ also works when some of the letters of v are barred.

It was proven in Proposition 2.7 that $\boldsymbol{F}_{\mathfrak{n}\mathfrak{F}}(\alpha)$ satisfies (1)–(6). With the help of (7), these proofs may be extended to $\boldsymbol{F}_{\mathfrak{F}}(\alpha)$. For example, to prove (1), let v, w, x, y, z be normal flock words. For all x_i appearing in any of $v, \ldots, z$, use (7) to add the skew pair $x_i\overline{x}_i$ at the right side of each word $v, \ldots, z$. Then (1) holds for these words since their modifications are normal near flock words. Use (7) again to wipe out all skew pairs, returning $v, \ldots, z$ to satisfy (1). □

The proof of the next theorem builds on that for free near flocks.

Theorem 3.4. *For any non-zero cardinal α, $\boldsymbol{F}_{\mathfrak{F}}(\alpha)$ is the free flock on α generators.*

Proof. It was just proven that $\boldsymbol{F}_{\mathfrak{F}}(\alpha)$ is a flock. The argument that $\boldsymbol{F}_{\mathfrak{F}}(\alpha)$ satisfies the universal property for freedom is like that for Theorem 2.8. □

Proposition 3.5.

(*a*) *For each term t of type $\langle 3, 1\rangle$, there is a unique normal flock word w such that $\mathsf{F} \vdash t \approx w^\beta$.*

(*b*) *For any normal flock words v and w, $\mathsf{F} \vdash v^\beta \approx w^\beta$ iff $v = w$.*

Proof. By Proposition 2.9 there is a unique normal free near flock word w such that $\mathsf{nF} \vdash t \approx v^\beta$. Eliminating the semilattice part of v give w. □

Corollary 3.6. *The equational theory of flocks is decidable.*

Proof. Like that of Corollary 2.10. □

The proofs and structure of normal forms suggest building any free near flock as a subalgebra of a product of a free flock and a free semilattice. To set the stage, here is a sort review of semilattices. They are traditionally binary algebras with one operation $\vee$ that is idempotent, commutative and associative. A term-equivalent variety with a ternary operation $[,,]$ and a unary operation $^-$ is obtained by 'stammering' the binary operation, and making the unary a dummy:

$$[x, y, z] = (x \vee y) \vee z,$$
$$\overline{x} = x.$$

An example of a semilattice lies in the semilattice parts w^σ of normal near flock words w. They make up the free semilattice $\boldsymbol{F}_{\mathfrak{s}\mathfrak{L}}(\alpha)$ on α generators $x_i\overline{x}_i$ $(i < \alpha)$. The ternary operation $[u^\sigma, v^\sigma, w^\sigma]$ is the word consisting of all skew pairs $x_i\overline{x}_i$ occurring in any of u^σ, v^σ or w^σ, arranged in order of ascending index, that is, $[u^\sigma, v^\sigma, w^\sigma] = u^\sigma \Cup v^\sigma \Cup w^\sigma$. Bar does nothing. $\boldsymbol{F}_{\mathfrak{s}\mathfrak{L}}(\alpha)$ is term-equivalent to the semilattice of all nonempty finite subsets of a set with α elements. It is almost a distributive lattice in that every interval of it is a distributive lattice with the operations of union and intersection. All that is missing to make it distributive is the empty set, a bottom element.

Theorem 3.7. *For any nonzero cardinal α, the free near flock $\boldsymbol{F}_{\mathfrak{nF}}(\alpha)$ is isomorphic to a subalgebra of the product, $\boldsymbol{F}_{\mathfrak{F}}(\alpha) \times \boldsymbol{F}_{\mathfrak{sL}}(\alpha)$, of a corresponding free flock and free semilattice. The carrier of the subalgebra is $\{\langle w^{\phi}, w^{\sigma}\rangle \mid w \in F_{\mathfrak{nF}}(\alpha)\}$.*

Proof. Define a function $h : F_{\mathfrak{nF}}(\alpha) \to F_{\mathfrak{F}}(\alpha) \times F_{\mathfrak{sL}}(\alpha)$ by $h(w) = \langle w^{\phi}, w^{\sigma}\rangle$. That h is an injection follows from the definition of normal words. It is a homomorphism since it preserves the operations:

$$h(\overline{w}) = h\left(\overline{w^{\phi}w^{\sigma}}\right) = h\left(\overline{w^{\phi}}w^{\sigma}\right) = h\left(\overline{w}^{\phi}w^{\sigma}\right) = \langle \overline{w^{\phi}}, w^{\sigma}\rangle = \overline{\langle w^{\phi}, w^{\sigma}\rangle} = \overline{h(w)};$$

and $h([u,v,w] = [h(u), h(v), h(w)]$ similarly. $\square$

With this theorem we may check the top part of Figure 1.

Theorem 3.8.

(*a*) *The variety of near flocks is the join of those of flocks and semilattices:* $\mathfrak{nF} = \mathfrak{F} \vee \mathfrak{sL}$.

(*b*) *The variety of near flocks is the join of those of flocks and near heaps:* $\mathfrak{nF} = \mathfrak{F} \vee \mathfrak{nH}$.

Proof. (*a*). The inclusions of their defining identities are passed to the varieties themselves, and hence $\mathfrak{F} \vee \mathfrak{sL} \subseteq \mathfrak{nF}$. As each free near flock $\boldsymbol{F}_{\mathfrak{nF}}(\alpha)$ is a subalgebra of a product of a flock and a semilattice (Theorem 3.7), we have that $\boldsymbol{F}_{\mathfrak{nF}}(\alpha)$ belongs to the join $\mathfrak{F} \vee \mathfrak{sL}$. As any near flock $\boldsymbol{A}$ is a homomorphic image of a free near flock, it follows that $\boldsymbol{A}$ is in the join. Therefore, $\mathfrak{nF} \subseteq \mathfrak{F} \vee \mathfrak{sL}$.

(*b*). This follows from (*a*) by the inclusion of varieties: $\mathfrak{sL} \subseteq \mathfrak{nH} \subseteq \mathfrak{nF}$. $\square$

In the language of extensions and Płonka sums more can be said about the structure of near flocks. We first define extensions and prove Theorem 3.10. Then we define connecting homomorphisms that turn this extension into a Płonka sum.

Definition 3.9. An *extension* (or *union* or *sum*) of a nonempty set $\mathcal{A}$ of algebras by another algebra $\boldsymbol{B}$ (all of the same type) is an algebra $\boldsymbol{E}$ and a congruence θ of $\boldsymbol{E}$ such that:

1. each congruence class of θ that is an algebra is isomorphic to a member of $\mathcal{A}$;

2. each member of $\mathcal{A}$ is isomorphic to some congruence class of θ; and

3. $\boldsymbol{E}/\theta$ is isomorphic to $\boldsymbol{B}$.

This definition came from specializing Mal'cev's definition for classes of algebras to individual algebras [11]. In turn, his definition grew out of classical extensions in group theory, where not every coset is a subgroup. However, when $\mathbf{B}$ is idempotent, say a semilattice, then all the congruence classes of $\mathbf{E}/\theta$ will be subalgebras.

Theorem 3.10. *Each near flock $\boldsymbol{A}$ is an extension of flocks by a semilattice.*

The proof of this theorem proceeds by a series of lemmas and interspersed definitions. Easy proofs are omitted without mention.

Now assume that $\boldsymbol{A}$ is a near flock. A congruence θ of $\boldsymbol{A}$ is found such that $\boldsymbol{A}$ is an extension of its congruence classes a/θ by its quotient $\boldsymbol{A}/\theta$.

Definition 3.11. On A define the binary relation:

$$a \leqslant b \quad \text{if} \quad [a, \overline{a}, b] = b. \tag{15}$$

Lemma 3.12.
(*a*) *The relation $\leqslant$ is a quasi-order.*
(*b*) *The operations $^-$ and $[,,]$ preserve $\leqslant$.*

Proof. (*a*). Reflexivity is clear from (5). To prove transitivity, suppose that $a \leqslant b$ and $b \leqslant c$, that is, $[a, \overline{a}, b] = b$ and $[b, \overline{b}, c] = c$. Then by (1),

$$[a, \overline{a}, c] = [a, \overline{a}, [b, \overline{b}, c]] = [[a, \overline{a}, b], \overline{b}, c] = [b, \overline{b}, c] = c,$$

and hence $a \leqslant c$.

(*b*). Bar is preserved by (3). We prove that $\leqslant$ preserves $[,,]$ in its middle argument; the other arguments are simpler.

We assume that $b \leqslant d$, that is $[b, \overline{b}, d] = d$, and prove that $[a, b, c] \leqslant [a, d, c]$, with the help of (1), (2), (5), (10) and (12):

$$\begin{aligned} [[a,b,c], \overline{[a,b,c]}, [a,d,c]] &= [a, \overline{a}, [b, \overline{b}, [c, \overline{c}, [a,d,c]]]] = [a, \overline{a}, [b, \overline{b}, [a, d, [c, \overline{c}, c]]]] \\ &= [a, \overline{a}, [b, \overline{b}, [a,d,c]]] = [b, \overline{b}, [a,d,c]] \\ &= [a, d, [b, \overline{b}, c] = [a, [b, \overline{b}, d], c] \\ &= [[a, d, c]. \end{aligned}$$

□

Definition 3.13. On A define the binary relation θ by:

$$a \,\theta\, b \quad \text{if} \quad a \leqslant b \text{ and } b \leqslant a.$$

Lemma 3.14. *The relation θ is a congruence of $\boldsymbol{A}$.*

Proof. By Lemma 3.12, $\leqslant$ is a quasi-order preserving $^-$ and $[,,]$. Therefore, θ is an equivalence relation preserving the operations. □

Lemma 3.15. *Each coset of θ is a flock.*

Proof. First one must prove that, for any element of e of A, the coset e/θ is an algebra, that is, it is closed to the operations $^-$ and $[,,]$. To prove closure to $^-$, suppose $a \in e/\theta$. Then $a \,\theta\, e$, and hence from the definition of θ,

$$[a, \overline{a}, e] = e \quad \text{and} \quad [e, \overline{e}, a] = a.$$

From the first equation, with the identities for near flocks we get that $[\overline{a}, \overline{\overline{a}}, e] = [\overline{a}, a, e] = [a, \overline{a}, e] = e$, and so $\overline{a} \leqslant e$. From the second, similarly $e \leqslant \overline{a}$, and thus $\overline{a} \in e/\theta$.

To prove closure to $[,,]$, suppose $a, b, c \in e/\theta$. As before, $[a, \overline{a}, e] = e$ and $[e, \overline{e}, a] = a$, and likewise for b and c. From these equations, Proposition 2.1, and the axioms for near flocks, we deduce that

$$\begin{aligned}[][a,b,c], \overline{[a,b,c]}, e] &= [[a,b,c], [\overline{a}, \overline{b}, \overline{c}], e] = [[[a,b,c], \overline{c}, \overline{b}], \overline{a}, e] \\ &= [[a,b,\overline{b}], \overline{a}, [c, \overline{c}, e]] = [[a,b,\overline{b}], \overline{a}, e] \\ &= [a, \overline{a}, [b, \overline{b}, e]] = [a, \overline{a}, e] = e.\end{aligned}$$

Hence $[a, b, c] \leqslant e$, and with less work $e \leqslant [a, b, c]$; therefore, $[a, b, c] \in e/\theta$.

To show that e/θ is a flock one need only show that (7) is an identity in e/θ; that is, show $[a, \overline{a}, b] = b$ for all a and b related by θ; but this last implies $a \leqslant b$, that is, $[a, \overline{a}, b] = b$. □

Lemma 3.16. *The quotient $\boldsymbol{A}/\,\theta$ is a semilattice.*

Proof. We need only prove (8) and (9) in $\boldsymbol{A}/\,\theta$. For the latter, this amounts to showing that $[a, a, b]\,\theta\,[a, b, b]$ for any a and b in A. Arguing with the axioms and Proposition 2.1 as before, we show that $[[a, a, b], \overline{[a, a, b]}, [a, b, b]] = \cdots = [a, b, b]$, which implies $[a, a, b] \leqslant [a, b, b]$. The converse of this relation is proven similarly, and so the two sides are related by θ. This completes the proof of Theorem 3.10. □

This extension is refined with Płonka sums [13], which are defined here only for near flocks. In [7, Theorem 11] a Płonka sum of heaps is also called a 'strong semilattice of heaps'. We need the partial order found in any semilattice $\boldsymbol{S}$:

$$r \leqslant s \text{ if } [r, \overline{r}, s] = s \quad (r, s \in S).$$

Definition 3.17. A near flock $\boldsymbol{A}$ is a *Płonka sum* of flocks if it is the union of a family $\{\boldsymbol{A}_s \mid s \in S\}$ of disjoint flocks indexed by a semilattice $\boldsymbol{S}$ together with a family of homomorphisms, $\{h_{rs} : \boldsymbol{A}_r \to \boldsymbol{A}_s \,|\, r \leqslant s \text{ in } \boldsymbol{S}\}$, that evaluate the ternary and unary operations of $\boldsymbol{A}$:

$$[a,b,c]^{\boldsymbol{A}} = [(h_{\pi(a),s}(a), h_{\pi(b),s}(b), h_{\pi(c),s}(c)]^{\boldsymbol{A}_s}, \text{ where } s = [\pi(a), \pi(b), \pi(c)]^{\boldsymbol{S}}; \tag{16}$$

$$\overline{(a)}^{\boldsymbol{A}} = \overline{h_{\pi(a),s}(a)}^{\boldsymbol{A}_s}, \qquad \text{where } s = \overline{\pi(a)}^{\boldsymbol{S}}. \tag{17}$$

Here the homomorphisms are assumed to be functorial in that $h_{st} \circ h_{rs} = h_{rt}$ when $r \leqslant s \leqslant t$; and π is the projection map from the disjoint flocks to their indices: $\pi(a) = s$ if $a \in A_s$. The class of Płonka sums of flocks is denoted $s_P\mathfrak{F}$.

The next theorem follows from the more general theory of Płonka [15, Theorem 7.1]. It is also proven in [7, Section 4]; but we sketch another proof that depends in part on Theorem 3.10.

Theorem 3.18. *Every Płonka sum of flocks is a near flock, and every near flock is a Płonka sum of flocks. In short,* $\mathfrak{n}\mathfrak{F} = s_P\mathfrak{F}$.

Proof. That a Płonka sum of flocks is a near flock follows from proving that the identities satisfied by a Płonka sum $\boldsymbol{A}$ are precisely those common to the stalks $\boldsymbol{A}_s$ and semilattices. This follows from verifying by induction on terms that for any term t with n variables,

$$t^{\boldsymbol{A}}(a_1,\ldots,a_n) = t^{\boldsymbol{A}_s}(h_{s_1,s}(a_1),\ldots,h_{s_n,s}(a_n)) \qquad (a_i \in A_{s_i});$$

here s is the semilattice join of the s_i.

For the other direction, by Theorem 3.10, $\boldsymbol{A}$ is the extension of flocks by a semilattice. Let θ be the congruence in Definition 3.13. To create a Płonka sum take the index set S to be the set A/θ of congruence classes a/θ of θ, and define the projection, $\pi(a) = a/\theta$. Define the connecting homomorphisms by

$$h_{\frac{a}{\theta},\frac{b}{\theta}}(x) = [x,\overline{b},b] \qquad (a,b \in A \text{ and } x \in a/\theta). \tag{18}$$

It remains to be proven that these homomorphisms are well-defined and functorial, and that the operations are evaluated correctly.

These connecting homomorphisms are well-defined since different choices of related a's and related b's yield the same answer for (18). In detail, supposing $x \in a_1/\theta$, $a_1\theta a_2$ and $b_1\theta b_2$, we know that $b_1 = [b_1,\overline{b}_2,b_2]$ and find that

$$h_{\frac{a_1}{\theta},\frac{b_1}{\theta}}(x) = [x,\overline{b}_1,b_1] = [x,\overline{b}_1,[b_1,\overline{b}_2,b_2]] = [x,[\overline{b}_2,b_1,\overline{b}_1],b_2] = [x,\overline{b}_2,b_2] = h_{\frac{a_2}{\theta},\frac{b_2}{\theta}}(x).$$

They are functorial since, if $a \leqslant b \leqslant c$ and $x \in a/\theta$, then

$$h_{\frac{b}{\theta},\frac{c}{\theta}}(h_{\frac{a}{\theta},\frac{b}{\theta}}(x)) = h_{\frac{b}{\theta},\frac{c}{\theta}}([x,\overline{b},b]) = [[x,\overline{b},b],\overline{c},c] = [x,[\overline{c},b,\overline{b}],c] = [x,\overline{c},c] = h_{\frac{a}{\theta},\frac{c}{\theta}}(x).$$

They evaluate correctly according to (16) and (17) since for a,b,c in A and $d = [a,b,c]$, we have in $\boldsymbol{A}/\theta$ that

$$s = [\pi(a),\pi(b),\pi(c)] = \left[\frac{a}{\theta},\frac{b}{\theta},\frac{c}{\theta}\right] = \frac{[a,b,c]}{\theta} = \frac{d}{\theta},$$

and hence, with the help of (13) and (5),

$$\begin{aligned}
[(h_{\pi(a),s}(a),h_{\pi(b),s}(b),h_{\pi(c),s}(c)] &= [h_{\frac{a}{\theta},\frac{d}{\theta}}(a)]h_{\frac{b}{\theta},\frac{d}{\theta}}(b)]h_{\frac{c}{\theta},\frac{d}{\theta}}(c)] \\
&= [[a,\overline{d},d],[b,\overline{d},d],[c,\overline{d},d]] \\
&= [[a,b,c],\overline{d},d] \\
&= [[a,b,c],\overline{[a,b,c]},[a,b,c]] \\
&= [a,b,c].
\end{aligned}$$

For the unary operation, since $s = \overline{\pi(a)}^{\boldsymbol{S}} = \pi(a) = \frac{a}{\theta}$, we check that $\overline{h_{\pi(a),s}(a)} = \overline{h_{\frac{a}{\theta},\frac{a}{\theta}}(a)} = \overline{[a,\overline{a},a]} = \overline{a}$. $\square$

A free near flock may also be described as a Płonka sum; this is a refinement of Theorem 3.7. We realize this by looking closely at the definition of $F_{\mathfrak{nF}}(\alpha)$.

Theorem 3.19. *For a nonzero cardinal α, the free near flock $F_{\mathfrak{nF}}(\alpha)$ is the Płonka sum of the free flocks $F_{\mathfrak{F}}(w)$ indexed by elements w of the free semilattice $F_{\mathfrak{sL}}(\alpha)$. Here the $F_{\mathfrak{F}}(w)$ are free flocks on generators that are the skew pairs $x_i\overline{x}_i$ in w.*

The article [17] also describes free near flocks as Płonka sums of flocks, but assumes the free flocks $F_{\mathfrak{F}}(w)$ are already known.

The lattice of subvarieties of $\mathfrak{nF}$ has been described in [5] and [18, Section 4.3].

Theorem 3.20. *The lattice of varieties of near flocks is isomorphic to the product of the lattice of varieties of flocks and the two-element lattice.A subvariety of $\mathfrak{nF}$ is either a subvariety $\mathfrak{K}$ of $\mathfrak{F}$, or a join, $\mathfrak{K} \vee \mathfrak{sL}$, of it with the variety of semilattices.*

There is a curiosity about the flocks $\langle A; [\,,\,] \rangle$ originally defined by (1), (2) and the unique solvability of $[\,,\,]$. The class $\mathfrak{F}_3$ of all such is categorically isomorphic to $\mathfrak{F}$. However, $\mathfrak{F}$ is a variety, but $\mathfrak{F}_3$ is not. To understand this, let A be the set $\{x_0^n \mid n \geq 1, n \text{ odd}\}$ of words in $F_{\mathfrak{F}}(1)$. This set is closed to $[\,,\,]$ and thus it is a subalgebra of $\langle F_{\mathfrak{F}}(1); [\,,\,] \rangle$. Because there is no bar, the operation $[\,,\,]$ is not solvable in it. Hence $\mathfrak{F}_3$ is not closed to taking subalgebras, and so fails to be a variety. Therefore, $\mathfrak{F}_3$ is not definable by identities.

4. Near heaps

The variety $\mathfrak{nH}$ of near heaps is the class of algebras satisfying the identities: (1)–(6) and (8). The last identity means that the bar operation may be omitted, and we will do so for the remainder of this article, changing the type of near heaps, heaps and semilattices from $\langle 3, 1 \rangle$ to $\langle 3 \rangle$, only retaining the ternary operation $[\,,\,]$. With that understanding, the defining identities are equivalent to

$$[v,w,[x,y,z]] \approx [[v,w,x],y,z], \tag{1}$$
$$[v,w,[x,y,z]] \approx [v,[y,x,w],z], \tag{2}$$
$$[x,x,x] \approx x, \tag{19}$$
$$[x,x,y] \approx [y,x,x], \tag{20}$$

which is the way Hawthorn and Stokes [7] introduced near heaps.

These identities hold in any group when $[x,y,z]$ is interpreted as $x(y^{-1}z)$, and in any semilattice when $[x,y,z]$ is interpreted as $x(yz)$. In this section a new normal form describes the elements of free near heaps. In the next, the variety of near heaps is proven to be the join of the varieties of heaps and semilattices, in fact, Płonka sums.

Definition 4.1. *Near heap words* w now have no bars, they are simply finite sequences of letters x_i. With bars no more, twin pairs take the place of skew pairs; a *twin pair* is a double occurrence $x_i x_i$ of the same letter adjacent to itself. A *normal near heap word* is in two parts: on the left will be the heap part $w^{\widehat{\phi}}$, a string of odd length of isolated occurrences of individual letters; on the right will be the semilattice part $w^{\widehat{\sigma}}$, a string of twin pairs, one for each letter occurring in w, and ordered by increasing indices, with no twin pair duplicated. Hats over operators indicate their adjustment to there no longer being any bars. Recall that w^{β} restores brackets, associated to the left.

A normal near heap word w is derived from nH for any term t built from the ternary operation and letters alone. To convert t use (1) and (2) to associate all the brackets to the left; (2) may change the order of the letters. Move to the right side any twin pair by using (10) and (11). (They have no bars now.) Reorder these pairs by increasing indices, removing duplicate pairs with (5). If some letter is isolated on the left side and does not occur on the right, triplicate it with (19) to create a twin pair, and move the twin pair to the side, absorbing it among the twin pairs already ordered there.

For example, if t is $[x_3, x_2, [x_2, x_1, x_3]]$, then, with the algorithm in the proof of the next lemma, w is $x_3x_1x_3\, x_1x_1x_2x_2x_3x_3$ with $w^{\widehat{\phi}} = x_3x_1x_3$ and $w^{\widehat{\sigma}} = x_1x_1x_2x_2x_3x_3$.

Definition 4.2. For any cardinal α, the algebra $\boldsymbol{F}_{\mathfrak{n}\mathfrak{H}}(\alpha)$ has as carrier $F_{\mathfrak{n}\mathfrak{H}}(\alpha)$ of all normal near heap words with letters from $\{x_i \mid i < \alpha\}$ and a ternary operation defined by,

$$[u, v, w] = (u^{\widehat{\phi}} v^{\widehat{\phi}\rho} w^{\widehat{\phi}})^{\widehat{\phi}} (u^{\widehat{\sigma}} \uplus v^{\widehat{\sigma}} \uplus w^{\widehat{\sigma}}),$$

where $u^{\widehat{\sigma}} \uplus v^{\widehat{\sigma}} \uplus w^{\widehat{\sigma}}$ is the sequence of all twin pairs in order of increasing index without repetition.

Proposition 4.3. *For any cardinal α, $\boldsymbol{F}_{\mathfrak{n}\mathfrak{H}}(\alpha)$ is a near heap.*

Proof. When the operation, $\overline{x} = x$, is introduced into the type, the identities of nH follow readily from those of nF. □

Theorem 4.4. *For any cardinal α, $\boldsymbol{F}_{\mathfrak{n}\mathfrak{H}}(\alpha)$ is the free near heap on α generators.*

Proof. We use the universal mapping property of free algebras. Let $\boldsymbol{A}$ be a near heap generated by $\{a_i \mid i < \alpha\}$. Define h on a w of $F_{\mathfrak{n}\mathfrak{H}}(\alpha)$ in the letters $x_{i_1}, x_{i_2}, \ldots, x_{i_n}$ by $h(w) = w^{\beta}(a_{i_1}, a_{i_2}, \ldots, a_{i_n})$. Proving that h is the unique homomorphism of $\boldsymbol{F}_{\mathfrak{n}\mathfrak{H}}(\alpha)$ to $\boldsymbol{A}$ taking x_i to a_i parallels the proof of Theorem 2.8. □

Proposition 4.5.

(*a*) *For any near heap term t, there is a unique normal near heap word w such that $\mathsf{nH} \vdash t \approx w^{\beta}$.*

(*b*) *For any normal near heap words v and w, $\mathsf{nF} \vdash v^{\beta} \approx w^{\beta}$ iff $v = w$.*

Proof. Existence comes from the algorithm of Definition 4.1, which uses only the identities of nH and their consequences.

Uniqueness and part (b) parallel the proof of Proposition 2.9. $\square$

Corollary 4.6. *The equational theory of near heaps is decidable.*

Proof. Like that of Corollary 2.10. $\square$

5. Heaps

These were defined in Section as algebras satisfying the identities (1)–(8). As the last identity makes the bar $^{-}$ pointless, these identities are equivalent to (1), (2) and

$$[x, x, y] \approx y \approx [y, x, x] \tag{21}$$

in algebras with only a ternary operation. Let H be this last set of identities. It is proven that the variety $\mathfrak{nH}$ of near heaps is the smallest variety containing heaps and semilattices. Even better, any near heap is a Płonka sum of heaps over a semilattice. From results in the literature, the lattice of subvarieties of $\mathfrak{nH}$ is sketched, and their subdirectly irreducibles are determined modulo those of heaps. The equivalence of these varieties with the traditional ones for ordinary groups and semilattices with binary operations will be addressed in Section 6. Heaps were first studied by Prüfer [16] in the context of commutative groups where $[x, y, z] = x - y + z$.

Of all the free algebras in this article, free heaps are the simplest to describe; their elements are just the left part, the heap part, of normal near heap words.

Definition 5.1. A *normal heap word* is a string of letters of odd length in which no letter occurs next to itself. The set $F_{\mathfrak{H}}(\alpha)$ of normal heap words on the alphabet $\{_i|\ i < \alpha\}$ is the carrier of the algebra $\boldsymbol{F}_{\mathfrak{H}}(\alpha)$ with the ternary operation

$$[u, v, w] = (uv^{\rho}w)^{\widehat{\phi}}.$$

Proposition 5.2. *For α a nonzero cardinal, $\boldsymbol{F}_{\mathfrak{H}}(\alpha)$ is a heap.*

Proof. As a heap is a near heap, only axiom (21) needs be proven:

$$[v, v, w] = (vv^{\rho}w)^{\widehat{\phi}} = (x_{i_1}x_{i_2}\dots x_{i_n}x_{i_n}\dots x_{i_2}x_{i_1}w)^{\widehat{\phi}} = w,$$

when $v = x_{i_1}x_{i_2}\dots x_{i_n}$; here (7) cancels duplicate pairs successively. $\square$

Theorem 5.3. *For α a nonzero cardinal, $\boldsymbol{F}_{\mathfrak{H}}(\alpha)$ is the free heap on α generators.*

Proof. This parallels the proof for Theorem 2.8. $\square$

Proposition 5.4.

(a) *For any heap term t, there is a unique normal heap word such that* $\mathsf{H} \vdash t \approx w^{\beta}$.

(b) *For any normal heap words v and w,* $\mathsf{nF} \vdash v^{\beta} \approx w^{\beta}$ *iff* $v = w$.

Proof. This is like that of Proposition 3.5. □

Corollary 5.5. *The equational theory of heaps is decidable.*

Proof. Similar to that for Corollary 2.10. □

Theorem 5.6. *For any nonzero cardinal α, the free near heap on α generators is isomorphic to a subalgebra of the product of the free heap and the free semilattice, both on α generators. Symbolically, $\boldsymbol{F}_{\mathfrak{n}\mathfrak{H}}(\alpha) \hookrightarrow \boldsymbol{F}_{\mathfrak{H}}(\alpha) \times \boldsymbol{F}_{\mathfrak{s}\mathfrak{L}}(\alpha)$.*

Proof. Define the function $F_{\mathfrak{n}\mathfrak{H}}(\alpha) \hookrightarrow F_{\mathfrak{H}}(\alpha) \times F_{\mathfrak{s}\mathfrak{L}}(\alpha)$ by $h(w) = \langle w^{\widehat{\phi}}, w^{\widehat{\sigma}} \rangle$. It is a homomorphism since it preserves the ternary operation:

$$\begin{aligned} h([u,v,w]) &= \langle [u,v,w]^{\widehat{\phi}}, [u,v,w]^{\widehat{\sigma}} \rangle \\ &= \langle (u^{\widehat{\phi}} v^{\widehat{\phi}\rho} w^{\widehat{\phi}})^{\widehat{\phi}}, (u^{\widehat{\sigma}} v^{\widehat{\sigma}} w^{\widehat{\sigma}})^{\widehat{\sigma}} \rangle \\ &= \langle [u^{\widehat{\phi}}, v^{\widehat{\phi}}, w^{\widehat{\phi}}], [u^{\widehat{\sigma}}, v^{\widehat{\sigma}}, w^{\widehat{\sigma}}] \rangle \\ &= [\langle u^{\widehat{\phi}}, u^{\widehat{\sigma}} \rangle, \langle v^{\widehat{\phi}}, v^{\widehat{\sigma}} \rangle, \langle w^{\widehat{\phi}}, w^{\widehat{\sigma}} \rangle] \\ &= [h(u), h(v), h(w)]. \end{aligned}$$ □

Theorem 5.7. *The variety of hear heaps is the join of the varieties of heaps and semilattices: $\mathfrak{n}\mathfrak{H} = \mathfrak{H} \vee \mathfrak{s}\mathfrak{L}$; that is, it is the smallest variety containing them.*

Proof. It is the proof of Theorem 3.8 mutatis mutandis. □

Theorem 5.8. *The lattice of Figure 1 is a sublattice of the lattice of all varieties of algebra with one ternary operation and one unary operation.*

Proof. Note that the free algebras of the different varieties in Figure 1 being non-isomorphic shows that the inclusions in it are proper. That each join of Figure 1 is the smallest variety including those below it is covered by Theorems 3.8 and 5.7. For the each meet of the figure, recall that the meet of two varieties is their intersection, and that the inclusions of the varieties in Figure 1 correspond to that of their generating sets, for example, that $\mathsf{F} \cap \mathsf{nH} = \mathsf{H}$. □

The next theorem follows immediately from Theorem 3.10. It is also proven in [7, Section 4] in a different language, and also follows from [18, Theorem 4.3.2].

Theorem 5.9. *Every Płonka sum of heaps is a near heap, and every near heap is a Płonka sum of heaps. In short, $\mathfrak{n}\mathfrak{H} = s_P\mathfrak{H}$.*

In parallel with Theorem 3.19, a free near heap may also be described as a Płonka sum of free heaps over a free semilattice (see also [17] and [18, Theorem 4.3.8]).

6. Types for groups and heaps

This section clarifies the relationship between groups with a binary operation and heaps with a ternary one. We view their varieties as categories. Two functors pass back and forth between them, giving almost a categorical equivalence. To make clear what is preserved, the intermediary of pointed heaps is introduced. The point serves as an identity element and can be chosen arbitrarily in a heap. Some of these ideas and results were presented noncategorically by Baer [1] and Certaine [3], where there are many references to their origins. See [6] for related concepts.

Definition 6.1. The variety $\mathfrak{G}$ of *groups* is the class of all algebras $\langle G; \times, ^{-1}, e\rangle$ of type $\langle 2, 1, 0\rangle$ satisfying these identities:

$$x \times (y \times z) \approx (x \times y) \times z,$$
$$x \times x^{-1} \approx 1 \approx x^{-1} \times x,$$
$$1 \times x \approx x \approx x \times 1.$$

The variety $\mathfrak{pH}$ of *pointed heaps* consists of all algebras $\langle G; [\,,\,,\,], e\rangle$ of type $\langle 3, 0\rangle$ satisfying identities (1)–(2) and (21).

Surprisingly, no additional identities beyond these defining heaps are needed to define pointed heaps. Identities cannot nail the constant e — its choice is arbitrary!

The varieties $\mathfrak{G}$ and $\mathfrak{pH}$ are term-equivalent and hence categorically isomorphic. To see this, replace the three operations $x \times y$, x^{-1} and 1 in a group by the two operations $[x, y, z] = x \times y^{-1} \times z$ and $e = 1$ in a pointed heap, and replace the operations $[x, y, z]$ and e in a pointed heap by $x \times y = [x, e, y]$, $x^{-1} = [e, x, e]$, and $1 = e$ in a group. Now we define an adjoint situation between $\mathfrak{pH}$ and $\mathfrak{G}$.

Definition 6.2. The function $D : \mathfrak{pH} \to \mathfrak{H}$ drops the constant e as an operation from any pointed heap $\langle A; [\,,\,,\,], e\rangle$. Homomorphisms are left alone by $D(h) = h$, although there may be more of them in $\mathfrak{H}$. The function $E : \mathfrak{H} \to \mathfrak{pH}$ uses the axiom of choice to add to each heap $\langle A; [\,,\,,\,]\rangle$ an arbitrary element e of A. A homomorphism $h : \boldsymbol{A} \to \boldsymbol{A}'$ in $\mathfrak{H}$ is mapped by E to one in $\mathfrak{pH}$ by the formula:

$$E(h)(a) = [h(a), h(e), e'] \qquad (a \in A),$$

where e and e' are the constants chosen by E.

Theorem 6.3. *The functions $D : \mathfrak{pH} \to \mathfrak{H}$ and $E : \mathfrak{H} \to \mathfrak{pH}$ are functors, and D is both a right and left adjoint of E.*

Proof. That D and E are indeed functors is straightforward to verify.

To show that D is a left adjoint of E, it is easiest to prove an equivalent universal situation: for all $\boldsymbol{A}$ in $\mathfrak{H}$, there exists an $\boldsymbol{B}$ in $\mathfrak{pH}$ and a homomorphism

$f : \boldsymbol{A} \to D(\boldsymbol{B})$ such that for all $\boldsymbol{B}'$ in $\mathfrak{p}\mathfrak{H}$ and all homomorphisms $h : \boldsymbol{A} \to D(\boldsymbol{B}')$ there is a unique homomorphism $\bar{h} : \boldsymbol{B} \to \boldsymbol{B}'$ such that this diagram commutes:

$$\begin{tikzcd}
\boldsymbol{A} \arrow[r, "f"] \arrow[d, "h"'] & D(\boldsymbol{B}) \arrow[d, "D(\bar{h})"] & \boldsymbol{B} \arrow[d, "\bar{h}"] \\
D(\boldsymbol{B}') \arrow[r, equal] & D(\boldsymbol{B}') & \boldsymbol{B}'
\end{tikzcd}$$

It follows that D is a left adjoint of E by Theorem 27.3 of [8].

It is proven similarly that D is a right adjoint of E. □

How far this adjunction falls short of a categorical equivalence is seen in a proposition that traces how common concepts pass across. Its proof is routine. But its statement needs the Cayley representation of elements of a group as permutations.

Definition 6.4. For any pointed heap $\boldsymbol{A}$, $\operatorname{Cay}\boldsymbol{A} = \{f_{ab} \mid a, b \in A\}$, where f_{ab} is the function given by $f_{ab}(c) = [a, b, c]$.

In the following, $\operatorname{Sub}\boldsymbol{A}$, $\operatorname{Con}\boldsymbol{A}$ and $\operatorname{Aut}\boldsymbol{A}$ mean respectively the sets of all subalgebras, congruences and automorphisms of an algebra $\boldsymbol{A}$. For sets $\mathcal{F}_1$ and $\mathcal{F}_2$ of functions, $\mathcal{F}_1 \circ \mathcal{F}_2$ means complex composition: $\{f_1 \circ f_2 \mid f_i \in \mathcal{F}_i\}$.

Proposition 6.5. *For $\boldsymbol{A}, \boldsymbol{A}_1, \boldsymbol{A}_2$ in $\mathfrak{p}\mathfrak{H}$ and their images $D\boldsymbol{A}, D\boldsymbol{A}_1, D\boldsymbol{A}_2$ in $\mathfrak{H}$:*

(1) Sub *$(D\boldsymbol{A})$ = the set of congruence classes of $\boldsymbol{A}$;*

(2) Con *$(D\boldsymbol{A})$* = Con *$\boldsymbol{A}$;*

(3) Aut *$(D\boldsymbol{A})$* = Aut $\boldsymbol{A}$ ∘ Cay $\boldsymbol{A}$ = Cay $\boldsymbol{A}$ ∘ Aut *$\boldsymbol{A}$;*

(4) Hom *$(D\boldsymbol{A}_1, D\boldsymbol{A}_2)$* = Cay $\boldsymbol{A}_2$ ∘ Hom *$(\boldsymbol{A}_1, \boldsymbol{A}_2)$;*

(5) *$D\boldsymbol{A}_1 \times D\boldsymbol{A}_2 = D(\boldsymbol{A}_1 \times \boldsymbol{A}_2)$.*

Since the congruences are the same under D, so are the simple algebras and subdirectly irreducibles.

Dudek [6, Section 4] approaches the groups in flocks by looking at the binary operations, $x \cdot_a y = [x, a, y]$, which are isomorphic groups in a given flock.

Płonka sums of groups are developed in [14].

7. Transfer of identities

What do the identities defining a variety of ordinary groups become when transfered to a corresponding variety of near heaps? This section starts with algorithms for modifying identities to define varieties of a new type, then a theorem justifies them, and two examples follow. There are two or three steps, depending on whether the subvariety is only a variety of heaps or it is a join of one with semilattices (Theorem 3.20). Notation is from Section 6. Regularity is needed.

Definition 7.1. An identity is *regular* if each variable occurring in a term on one side of it occurs also in the term on the other side. A variety is *regular* if it can be defined by regular identities. The *regularization* of a variety $\mathfrak{K}$ is the variety defined by the regular identities satisfied by $\mathfrak{K}$.

Step 1 — from $\mathfrak{G}$ to $\mathfrak{pH}$. Here is the recipe for the first step to translate a term t of type $\langle 2,1,0\rangle$ to one of type $\langle 3,0\rangle$; it follows the scheme in Section 6.

- Replace each product $t_1 \times t_2$ of subterms t_1 and t_2 of t by $[t_1, e, t_2]$.
- Replace each inverse t_1^{-1} of a subterm t_1 of t by $[e, t_1, e]$.
- Replace the constant 1 by e.

Write $\bar{t}$ for the translated term, and $\bar{t}_1 \approx \bar{t}_2$ for the translation of an identity $t_1 \approx t_2$. For a set K of identities defining a variety of groups, let $\overline{\mathsf{K}}$ be the set of translations.

Step 2 — from $\mathfrak{pH}$ to $\mathfrak{H}$. Assume w is a variable not in any of the identities defining $\mathcal{H}$, a subvariety of $\mathfrak{pH}$. Replace the constant e by w in all the identities of K. Write $\overline{\mathsf{K}}$ for the set modified identities.

Step 3 — from $\mathfrak{H}$ to $\mathfrak{sH}$. If a subvariety of $\mathfrak{nH}$ is the join of a subvariety $\mathfrak{K}$ of $\mathfrak{H}$ with $\mathfrak{sL}$, then the identities K defining $\mathfrak{K}$ must be regularized. One can do this for an identity, $t_1 \approx t_2$, by adding to the right side of the term t_1 the pair xx for any variable x that appears only in t_2 to get $[t_1, x, x] \approx t_2$, and likewise for t_2 .

Theorem 7.2.

(1) *For $\mathfrak{K}$ a subvariety of $\mathfrak{G}$ defined by a set K of identities, the set $\overline{\mathsf{K}}$ of term-translated identities of* Step 1 *defines $\overline{\mathfrak{K}}$, the subvariety of $\mathfrak{pH}$ of pointed heaps term-equivalent to the groups of $\mathfrak{G}$.*

(2) *For a subvariety $\mathfrak{K}$ of $\mathfrak{pH}$ defined by a set K of identities, the set $\overline{\mathsf{K}}$ of translated identities of* Step 2 *defines the subvariety, $\overline{\mathfrak{K}} = D(\mathfrak{K})$, of $\mathfrak{H}$.*

(3) *For a subvariety $\mathfrak{K}$ of $\mathfrak{nH}$ with a defining set K of identities, a defining set of identities for its join, $\mathfrak{K} \vee \mathfrak{sL}$, with the variety of semilattices is given by the regularization $\overline{\mathsf{K}}$ of K, as done in* Step 3.

Proof. (1). This follows from term-equivalence of $\mathfrak{G}$ of $\mathfrak{pH}$.

(2). We must show that, if an identity $t_1 \approx t_2$ is satisfied by an algebra $\boldsymbol{A}$ of $\mathfrak{K}$, then its translation $\bar{t}_1 \approx \bar{t}_2$ is satisfied by $D(\boldsymbol{A})$, and conversely. Write the terms as $t_i(x_1, \ldots, x_n, e)$ where the t_i are of type $\langle 3\rangle$. Then the translated terms will be $t_i(x_1, \ldots, x_n, w)$ with w replacing e. We will show that $t_1(a_1, \ldots, a_n, b) = t_2(a_1, \ldots, a_n, b)$ for all $a_1, \ldots, a_n, b$ in A. Define an automorphism α of $\boldsymbol{A}$ by $\alpha(x) = [x, b, e]$. Then $\alpha(b) = e$. So

$$\begin{aligned}\alpha(t_1(a_1,\dots,a_n,b)) &= t_1(\alpha(a_1),\dots,\alpha(a_n),\alpha(b))\\ &= t_1(\alpha(a_1),\dots,\alpha(a_n),e)\\ &= t_2(\alpha(a_1),\dots,\alpha(a_n),e)\\ &= t_2(\alpha(a_1),\dots,\alpha(a_n),\alpha(b))\\ &= \alpha(t_2(a_1,\dots,a_n,b)).\end{aligned}$$

Therefore, $t_1(a_1,\dots,a_n,b) = t_2(a_1,\dots,a_n,b)$. The converse is proven by replacing w by e.

We have shown for any identity $t_1 \approx t_2$ and any algebra $\boldsymbol{A}$ of $\mathfrak{K}$ that

$$\boldsymbol{A} \models t_1 \approx t_2 \text{ iff } D(\boldsymbol{A}) \models \bar{t}_1 \approx \bar{t}_2.$$

This equivalence also applies to the sets of identities:

$$\mathfrak{K} \models \mathsf{K} \text{ iff } \overline{\mathfrak{K}} \models \overline{\mathsf{K}}.$$

Therefore, $\overline{\mathsf{K}}$ defines $\overline{\mathfrak{K}}$ since K defines $\mathfrak{K}$.

(3). Let $t_1 \approx t_2$ be an identity of $\mathfrak{K} \vee \mathfrak{sL}$. We must show that it is derivable from the regularization $\overline{\mathsf{K}}$ of K. As an identity satisfied by $\mathfrak{K} \vee \mathfrak{sL}$, $t_1 \approx t_2$ is satisfied by $\mathfrak{K}$, and so is derivable from K alone. Such a derivation is a sequence of identities, each one of which is either in K or derivable from previous ones using the rules of equational logic. Now regularize each identity in this derivation. This is a derivation from $\overline{\mathsf{K}}$ of the regularization $\bar{t}_1 \approx \bar{t}_2$ of the original identity, with the proviso that some new identities must be interpolated to accommodate instances of substitution in equation logic. □

Two examples illustrate this process. The identities of Definition 6.1, which define groups, translate to identities that are seen to be equivalent to (1), (2) and (21), which define heaps. If the binary commutative law, $x \times y \approx y \times x$, is added, it becomes

$$[x, w, y] \approx [y, w, x] \tag{22}$$

in the first two steps. This is already regular, and so Step 3 is not needed. Hence the join of semilattices and commutative groups is defined by (1),(2), (21) and (22).

Elementary 2-groups are defined by the identity, $x \times x \approx 1$. The first and second steps give $[x, w, x] \approx w$, and the third regularizes it:

$$[x, w, x] \approx [w, x, x]. \tag{23}$$

So the join of semilattices and 2-groups is defined by (1), (2), (21) and (23).

As 2-groups are commutative, it is an elementary exercise to show directly in the language of heaps that (22) follows from (23).

A note on the references. Some of the notions in this paper have an extensive literature reaching back more than a century. A sampling is included here, from which the reader may find more, as well as related concepts.

References

[1] **R. Baer**, *Zur Einführung des Scharbegriffs*, J. Reine Angew. Math. **160** (1929), 199 – 207.

[2] **J.R. Büchi and J.B. Wright**, *The theory of proportionality as an abstraction of group theory*, Math. Ann. **130** (1955), 102 – 108.

[3] **J. Certaine**, *The ternary operation* $(abc) = ab^{-1}c$ *of a group*, Bull. Amer. Math. Soc. **49** (1943), 869 – 877.

[4] **W. Dörnte**, *Untersuchungen über einen verallgemeinerten Gruppenbegriff*, Math. Z. **29** (1929), 1 – 19.

[5] **J. Dudek and E. Graczyńska**, *The lattice of varieties of algebras*, Bull. Acad. Polon. Sci. Sér. Sci. Math. **29** (1981), 337 – 340.

[6] **W.A. Dudek**, *Ternary quasigroups connected with the affine geometry*, Algebras Groups Geom. **16** (1999), 329 – 354.

[7] **I. Hawthorn and T. Stokes**, *Near heaps*, Comment. Math. Univ. Carolin. **52** (2011), no. 2, 163 – 175.

[8] **H. Herrlich and G.E. Strecker**, *Category theory*, volume 1 of Sigma Series in Pure Math., Heldermann Verlag, Berlin, second edition, 1979.

[9] **C. Hollings**, *Mathematics across the Iron Curtain*, vol. 41 of History of Mathematics, Amer. Math. Soc., Providence, RI, 2014. A history of the algebraic theory of semigroups.

[10] **M.V. Lawson**, *Generalized heaps, inverse semigroups and Morita equivalence*, Algebra Universalis **66** (2011), 317 – 330.

[11] **A.I. Mal′cev**, *Multiplication of classes of algebraic systems*, Sibirsk. Mat. Ž., **8** (1967), 346 – 365.

[12] **R. Padmanabhan and J. Płonka**, *Idempotent reducts of abelian groups*, Algebra Universalis **11** (1980), 7 – 11.

[13] **J. Płonka**, *On a method of construction of abstract algebras*, Fund. Math. **61** (1967), 183 – 189.

[14] **J. Płonka**, *On the sum of a direct system of universal algebras with nullary polynomials*, Algebra Universalis **19** (1984), 197 – 207.

[15] **J. Płonka and A. Romanowska**, *Semilattice sums*, Universal algebra and quasigroup theory (Jadwisin, 1989), vol. 19 of Res. Exp. Math., pp. 123 – 158, Heldermann, Berlin, 1992.

[16] **H. Prüfer**, *Theorie der Abelschen Gruppen*, Math. Z. **20** (1924), 165 – 187.

[17] **A. Romanowska**, *On free algebras in some equational classes defined by regular equations*, Demonstratio Math. **11** (1978), 1131 – 1137.

[18] **A.B. Romanowska and J.D.H. Smith**, *Modes*, World Scientific Publishing Co. Inc., River Edge, NJ, 2002.

[19] **A.K. Sushkevich**, *The theory of generalized groups*, Kharkov, Kiev, 1937.

[20] **Á. Szendrei**, *Idempotent reducts of abelian groups*, Acta Sci. Math. (Szeged), **38** (1976), 171 – 182.

[21] **V.V. Vagner**, *The theory of generalized heaps and generalized groups*, Mat. Sbornik N.S. **32(74)** (1953), 545 – 632.

Received October 3, 2014
Revised February 8, 2016

8412 Cherry Hills Drive, N.E.
Albuquerque, New Mexico, 87111, U.S.A.
E-mail: a.knoebel@member.ams.org