Heaps and trusses

Alberto Facchini
Università di Padova, Italy

NonCommutative Rings and their Applications, VIII

Lens, 28 August 2023

Heaps: an old notion

Heaps were already considered by:

(1) H. Prüfer, *Theorie der Abelschen Gruppen. I. Grundeigenschaften*, Math. Z. **20** (1924), 165–187,

and

(2) R. Baer, *Zur Einfhrung des Scharbegriffs*, J. Reine Angew. Math. **160** (1929), 199–207.

Heaps: an old notion

Heaps were already considered by:

(1) H. Prüfer, *Theorie der Abelschen Gruppen. I. Grundeigenschaften*, Math. Z. **20** (1924), 165–187,

and

(2) R. Baer, *Zur Einfhrung des Scharbegriffs*, J. Reine Angew. Math. **160** (1929), 199–207.

Trusses are a much more recent notion:

(3) T. Brzeziński, *Trusses: between braces and rings*, Trans. Amer. Math. Soc. **372** (2019), no. 6, 4149–4176.

What I will present you today appears in:

(4) M. J. Arroyo Paniagua and A. Facchini, *Heaps and trusses*, 2023, available in arXiv:2308.00527

What I will present you today appears in:

(4) M. J. Arroyo Paniagua and A. Facchini, *Heaps and trusses*, 2023, available in arXiv:2308.00527, and in

(5) T. Brzeziński, S. Mereta and B. Rybołowicz, *From pre-trusses to skew braces*, Publ. Mat. **66** (2022), no. 2, 683–714.

Ternary operations

A set X endowed with a ternary operation $p\colon X \times X \times X \to X$

Ternary operations

A set X endowed with a ternary operation $p\colon X \times X \times X \to X$ (no identity is required to be satisfied, at the moment).

Ternary operations

A set X endowed with a ternary operation $p\colon X \times X \times X \to X$ (no identity is required to be satisfied, at the moment).

The pairs (X, p) form a variety of algebras in the sense of Universal Algebra.

Ternary operations

A set X endowed with a ternary operation $p\colon X \times X \times X \to X$ (no identity is required to be satisfied, at the moment).

The pairs (X, p) form a variety of algebras in the sense of Universal Algebra. Their morphisms $f\colon (X, p) \to (X', p')$ are the mappings $f\colon X \to X'$ such that $p'(f(x), f(y), f(z)) = f(p(x, y, z))$ for every $x, y, z \in X$.

Ternary operations

A set X endowed with a ternary operation $p\colon X \times X \times X \to X$ (no identity is required to be satisfied, at the moment).

The pairs (X, p) form a variety of algebras in the sense of Universal Algebra. Their morphisms $f\colon (X, p) \to (X', p')$ are the mappings $f\colon X \to X'$ such that $p'(f(x), f(y), f(z)) = f(p(x, y, z))$ for every $x, y, z \in X$.

In particular, these algebras (X, p) are the objects of a category, whose initial object is the empty set $\emptyset$ (with its unique ternary operation), and whose terminal objects are the singletons (with their unique ternary operation). We will denote by $*$ any such algebra with one element.

Mal'tsev operations.

Let $p\colon X \times X \times X \to X$ be a ternary operation on the set X.

Mal'tsev operations.

Let $p\colon X \times X \times X \to X$ be a ternary operation on the set X. We say that p is a *Mal'tsev operation* if $p(x, x, y) = y$ and $p(x, y, y) = x$ for every $x, y \in X$.

Mal'tsev operations.

Let $p\colon X \times X \times X \to X$ be a ternary operation on the set X. We say that p is a *Mal'tsev operation* if $p(x, x, y) = y$ and $p(x, y, y) = x$ for every $x, y \in X$.

It will be often convenient to replace the ternary operation p on a set X with an indexed family $\{\, b_y \mid y \in X \,\}$ of binary operations $b_y\colon X \times X \to X$ defined by $b_y(x, z) = p(x, y, z)$ for every $x, y, z \in X$.

Mal'tsev operations.

Let $p\colon X \times X \times X \to X$ be a ternary operation on the set X. We say that p is a *Mal'tsev operation* if $p(x, x, y) = y$ and $p(x, y, y) = x$ for every $x, y \in X$.

It will be often convenient to replace the ternary operation p on a set X with an indexed family $\{\, b_y \mid y \in X \,\}$ of binary operations $b_y\colon X \times X \to X$ defined by $b_y(x, z) = p(x, y, z)$ for every $x, y, z \in X$. The family of binary operations b_y is indexed in X itself.

Mal'tsev operations.

Let $p\colon X \times X \times X \to X$ be a ternary operation on the set X. We say that p is a *Mal'tsev operation* if $p(x, x, y) = y$ and $p(x, y, y) = x$ for every $x, y \in X$.

It will be often convenient to replace the ternary operation p on a set X with an indexed family $\{\, b_y \mid y \in X \,\}$ of binary operations $b_y\colon X \times X \to X$ defined by $b_y(x, z) = p(x, y, z)$ for every $x, y, z \in X$. The family of binary operations b_y is indexed in X itself.

Correspondingly, we get a family of *magmas*

Mal'tsev operations.

Let $p\colon X \times X \times X \to X$ be a ternary operation on the set X. We say that p is a *Mal'tsev operation* if $p(x,x,y) = y$ and $p(x,y,y) = x$ for every $x, y \in X$.

It will be often convenient to replace the ternary operation p on a set X with an indexed family $\{\, b_y \mid y \in X \,\}$ of binary operations $b_y\colon X \times X \to X$ defined by $b_y(x,z) = p(x,y,z)$ for every $x, y, z \in X$. The family of binary operations b_y is indexed in X itself.

Correspondingly, we get a family of *magmas* (= sets with a binary operation) (X, b_y), which is again indexed in X itself.

Mal'tsev operations.

Let $p\colon X \times X \times X \to X$ be a ternary operation on the set X. We say that p is a *Mal'tsev operation* if $p(x, x, y) = y$ and $p(x, y, y) = x$ for every $x, y \in X$.

It will be often convenient to replace the ternary operation p on a set X with an indexed family $\{\, b_y \mid y \in X \,\}$ of binary operations $b_y\colon X \times X \to X$ defined by $b_y(x, z) = p(x, y, z)$ for every $x, y, z \in X$. The family of binary operations b_y is indexed in X itself.

Correspondingly, we get a family of *magmas* (= sets with a binary operation) (X, b_y), which is again indexed in X itself.

We will also often use the notations $[x, y, z]$ instead of $p(x, y, z)$, and $x \cdot_y z$ instead of $b_y(x, z)$.

Mal'tsev operations.

Lemma

A ternary operation p on a set X is a Mal'tsev operation if and only if, for the corresponding indexed family $\{ b_y \mid y \in X \}$ of binary operations, the element y is a two-sided identity of the magma (X, b_y) for every $y \in X$.

Mal'tsev operations.

Lemma

A ternary operation p on a set X is a Mal'tsev operation if and only if, for the corresponding indexed family $\{ b_y \mid y \in X \}$ of binary operations, the element y is a two-sided identity of the magma (X, b_y) for every $y \in X$.

Notice that in a magma, that is, a set with a not-necessarily associative operation, a two-sided identity, when it exists, is unique.

Commutative ternary operations.

What does “commutative” mean for a ternary operation p?

Commutative ternary operations.

What does "commutative" mean for a ternary operation p? For a binary operation $\cdot$ it means that $x \cdot z = z \cdot x$ for all $x, z \in X$. Now we have replaced the ternary operation p with a bunch of binary operations $\cdot_y$. Hence it is natural to say that a ternary operation p on X is commutative if and only if all binary operations $\cdot_y$, $y \in X$, are commutative.

Commutative ternary operations.

What does "commutative" mean for a ternary operation p? For a binary operation $\cdot$ it means that $x \cdot z = z \cdot x$ for all $x, z \in X$. Now we have replaced the ternary operation p with a bunch of binary operations $\cdot_y$. Hence it is natural to say that a ternary operation p on X is commutative if and only if all binary operations $\cdot_y$, $y \in X$, are commutative. Therefore:

A ternary operation p on a set X is *commutative* if $p(x, y, z) = p(z, y, x)$ for every $x, y, z \in X$.

Commutative ternary operations.

A ternary operation p on a set X is *commutative* if $p(x, y, z) = p(z, y, x)$ for every $x, y, z \in X$.

(This motivation sounds a little too naive... We will see later that for heaps it is possible to define a notion of commutator of congruences, and that a heap is commutative if and only if all its congruences commute.

Commutative ternary operations.

A ternary operation p on a set X is *commutative* if $p(x, y, z) = p(z, y, x)$ for every $x, y, z \in X$.

(This motivation sounds a little too naive... We will see later that for heaps it is possible to define a notion of commutator of congruences, and that a heap is commutative if and only if all its congruences commute. This is exactly the case of groups, for instance.
For groups G, it is possible to define the commutator $[M, N]$ of any two normal subgroups M, N, of G and a group G is commutative (abelian) if and only if $[M, N] = 1$ for all its normal subgroups M, N.)

Associative ternary operations

What does "associative" mean for a ternary operation p?

Associative ternary operations

What does "associative" mean for a ternary operation p?
For a binary operation $\cdot$ it means that $x \cdot (y \cdot z) = (x \cdot y) \cdot z$ for all $x, y, z \in X$. Now we have replaced the ternary operation p with a bunch of binary operations $\cdot_y$. Hence it is natural to say that a ternary operation p on X is associative if and only if $(x \cdot_y z) \cdot_w u = x \cdot_y (z \cdot_w u)$ for every $x, y, z, w, u \in X$.

Associative ternary operations

What does "associative" mean for a ternary operation p?
For a binary operation $\cdot$ it means that $x \cdot (y \cdot z) = (x \cdot y) \cdot z$ for all $x, y, z \in X$. Now we have replaced the ternary operation p with a bunch of binary operations $\cdot_y$. Hence it is natural to say that a ternary operation p on X is associative if and only if $(x \cdot_y z) \cdot_w u = x \cdot_y (z \cdot_w u)$ for every $x, y, z, w, u \in X$. Therefore:

Associative ternary operations

What does "associative" mean for a ternary operation p?
For a binary operation $\cdot$ it means that $x \cdot (y \cdot z) = (x \cdot y) \cdot z$ for all $x, y, z \in X$. Now we have replaced the ternary operation p with a bunch of binary operations $\cdot_y$. Hence it is natural to say that a ternary operation p on X is associative if and only if $(x \cdot_y z) \cdot_w u = x \cdot_y (z \cdot_w u)$ for every $x, y, z, w, u \in X$. Therefore:

A ternary operation p on a set X is *associative* if $p(p(x, y, z), w, u) = p(x, y, p(z, w, u))$ for every $x, y, z, w, u \in X$.

Associative ternary operations

What does "associative" mean for a ternary operation p?
For a binary operation $\cdot$ it means that $x \cdot (y \cdot z) = (x \cdot y) \cdot z$ for all $x, y, z \in X$. Now we have replaced the ternary operation p with a bunch of binary operations $\cdot_y$. Hence it is natural to say that a ternary operation p on X is associative if and only if $(x \cdot_y z) \cdot_w u = x \cdot_y (z \cdot_w u)$ for every $x, y, z, w, u \in X$. Therefore:

A ternary operation p on a set X is *associative* if $p(p(x, y, z), w, u) = p(x, y, p(z, w, u))$ for every $x, y, z, w, u \in X$.

In particular, for $y = w$, we get that if a ternary operation p on a set X is associative, then all the binary operations b_y are associative, i.e., that all the magmas (X, b_y), $y \in X$, are semigroups.

Heaps

A set X with an associative Mal'tsev operation $[-,-,-]$ is called a *heap*.

Heaps

A set X with an associative Mal'tsev operation $[-,-,-]$ is called a *heap*.

A mapping $f\colon (X,[-,-,-]) \to (X',[-,-,-])$ between two heaps is a *heap morphism* if

$$f([x,x',x'']) = [f(x), f(x'), f(x'')]$$

for every $x, x', x'' \in X$.

Heaps

A set X with an associative Mal'tsev operation $[-,-,-]$ is called a *heap*.

A mapping $f\colon (X,[-,-,-]) \to (X',[-,-,-])$ between two heaps is a *heap morphism* if

$$f([x,x',x'']) = [f(x),f(x'),f(x'')]$$

for every $x,x',x'' \in X$.

The category of heaps will be denoted by Heap. In Heap, the initial object is $\emptyset$ and the terminal object is $*$.

Heaps

Theorem

Let (X, p) be a non-empty heap. Then all the monoids (X, b_x), $x \in X$, are pair-wise isomorphic groups.

Heaps

Theorem

Let (X, p) be a non-empty heap. Then all the monoids (X, b_x), $x \in X$, are pair-wise isomorphic groups.

Any (X, b_x) is a group because the inverse of any $y \in X$ is $[x, y, x]$.

Heaps

Theorem

Let (X, p) be a non-empty heap. Then all the monoids (X, b_x), $x \in X$, are pair-wise isomorphic groups.

Any (X, b_x) is a group because the inverse of any $y \in X$ is $[x, y, x]$.

The group isomorphisms $(X, b_x) \to (X, b_y)$ are the mappings $\tau_x^y \colon (X, b_x) \to (X, b_y)$ defined by $\tau_x^y(z) = [z, x, y]$ for every $x, y, z \in X$.

Heaps

Theorem

Let (X, p) be a non-empty heap. Then all the monoids (X, b_x), $x \in X$, are pair-wise isomorphic groups.

Any (X, b_x) is a group because the inverse of any $y \in X$ is $[x, y, x]$.

The group isomorphisms $(X, b_x) \to (X, b_y)$ are the mappings $\tau_x^y \colon (X, b_x) \to (X, b_y)$ defined by $\tau_x^y(z) = [z, x, y]$ for every $x, y, z \in X$.

A subset S of a heap is a *subheap* if $[x, y, z] \in S$ for every $x, y, z \in S$.

Example 1 (Brzeziński, Trans. Amer. Math. Soc., 2019)

Our "3-dimensional Euclidean geometrical real space" of Newtonian Physics.

Example 1 (Brzeziński, Trans. Amer. Math. Soc., 2019)

Our "3-dimensional Euclidean geometrical real space" of Newtonian Physics. Its set E_3 of points doesn't have a natural group structure (or a vector-space structure): the sum of two points doesn't have a natural meaning.

Example 1 (Brzeziński, Trans. Amer. Math. Soc., 2019)

Our “3-dimensional Euclidean geometrical real space” of Newtonian Physics. Its set E_3 of points doesn't have a natural group structure (or a vector-space structure): the sum of two points doesn't have a natural meaning. But as soon as we fix a point (an *origin*), we can define an addition using the Parallelogram Rule, and we get an abelian group. In fact, we get a 3-dimensional vector space over the field of real numbers.

Example 1 (Brzeziński, Trans. Amer. Math. Soc., 2019)

Our “3-dimensional Euclidean geometrical real space” of Newtonian Physics. Its set E_3 of points doesn’t have a natural group structure (or a vector-space structure): the sum of two points doesn’t have a natural meaning. But as soon as we fix a point (an *origin*), we can define an addition using the Parallelogram Rule, and we get an abelian group. In fact, we get a 3-dimensional vector space over the field of real numbers. Hence E_3 does not have a natural group structure, if we want it we need the unnatural choice of an origin.

Example 1 (Brzeziński, Trans. Amer. Math. Soc., 2019)

Our “3-dimensional Euclidean geometrical real space” of Newtonian Physics. Its set E_3 of points doesn't have a natural group structure (or a vector-space structure): the sum of two points doesn't have a natural meaning. But as soon as we fix a point (an *origin*), we can define an addition using the Parallelogram Rule, and we get an abelian group. In fact, we get a 3-dimensional vector space over the field of real numbers. Hence E_3 does not have a natural group structure, if we want it we need the unnatural choice of an origin. But E_3 does have a natural heap structure: if $A, B, C \in E_3$, we can define $p(A, B, C)$ with the Parallelogram Rule, so that $A, B, C, p(A, B, C)$ are, orderly, the vertex of a parallelogram, and in this way we get a heap (E_3, p).

Example 2

We can fix any line in the space, getting a subheap E_1 of the previous example E_3, or any plane, getting a subheap E_2 of E_3.

Example 3

Fix any group G and define a ternary operation p on G setting $p(x, y, z) = xy^{-1}z$ for every $x, y, z \in G$.

Example 3

Fix any group G and define a ternary operation p on G setting $p(x, y, z) = xy^{-1}z$ for every $x, y, z \in G$. Then (G, p) is a heap.

Example 3

Fix any group G and define a ternary operation p on G setting $p(x, y, z) = xy^{-1}z$ for every $x, y, z \in G$. Then (G, p) is a heap.

Every non-empty heap is of this form, and there is a natural functor of the category of groups into the categories of heaps.

Example 3

Fix any group G and define a ternary operation p on G setting $p(x, y, z) = xy^{-1}z$ for every $x, y, z \in G$. Then (G, p) is a heap.

Every non-empty heap is of this form, and there is a natural functor of the category of groups into the categories of heaps. Nevertheless these two categories are not equivalent, for instance the category of heaps does not have a null object (the category of groups and the category of heaps are not equivalent categories also if we eliminate the empty heap from the objects of the category of heaps).

Normal subheaps

Lemma

The following conditions are equivalent for a non-empty subheap S of a heap X:

(a) *there exists $e \in S$ such that for every $x \in X$ and every $s \in S$ there exists $t \in S$ such that*

$$[x, e, s] = [t, e, x].$$

(b) *For every $x \in X$ and every $e, s \in S$ there exists $t \in S$ such that $[x, e, s] = [t, e, x]$.*

(c) $[[x, e, s], x, e] \in S$ *for every $x \in X$ and every $e, s \in S$.*

Normal subheaps

Lemma

The following conditions are equivalent for a non-empty subheap S of a heap X:

(a) *there exists $e \in S$ such that for every $x \in X$ and every $s \in S$ there exists $t \in S$ such that*

$$[x, e, s] = [t, e, x].$$

(b) *For every $x \in X$ and every $e, s \in S$ there exists $t \in S$ such that $[x, e, s] = [t, e, x]$.*

(c) $[[x, e, s], x, e] \in S$ *for every $x \in X$ and every $e, s \in S$.*

A subheap S of a heap X is said to be a *normal subheap* if it is non-empty and satisfies the equivalent conditions of the lemma.

Normal subheaps

Corollary

The following conditions are equivalent for a subset S of a heap H:
(a) *there exists $e \in S$ such that S is a normal subgroup of (X, b_e).*
(b) *S is non-empty and S is a normal subgroup of (X, b_e) for every $e \in S$.*
(c) *S is a normal subheap of X.*

A remark

Some care is necessary here. We have chosen our terminology in such a way that the empty set is a heap,

A remark

Some care is necessary here. We have chosen our terminology in such a way that the empty set is a heap, the empty subset is a subheap of every heap,

A remark

Some care is necessary here. We have chosen our terminology in such a way that the empty set is a heap, the empty subset is a subheap of every heap, but normal subheaps are non-empty by definition.

A remark

Some care is necessary here. We have chosen our terminology in such a way that the empty set is a heap, the empty subset is a subheap of every heap, but normal subheaps are non-empty by definition. As a consequence:

(a) Subheaps of a heap form a complete lattice (every intersection of subheaps is a subheap).

A remark

Some care is necessary here. We have chosen our terminology in such a way that the empty set is a heap, the empty subset is a subheap of every heap, but normal subheaps are non-empty by definition. As a consequence:

(a) Subheaps of a heap form a complete lattice (every intersection of subheaps is a subheap).

(b) Congruences on a heap form a complete lattice (every intersection of congruences is a congruence).

A remark

Some care is necessary here. We have chosen our terminology in such a way that the empty set is a heap, the empty subset is a subheap of every heap, but normal subheaps are non-empty by definition. As a consequence:

(a) Subheaps of a heap form a complete lattice (every intersection of subheaps is a subheap).

(b) Congruences on a heap form a complete lattice (every intersection of congruences is a congruence).

(c) Normal heaps of a heap do not form a lattice in general, but only a partially ordered set, because the intersection of two normal subheaps can be empty.

A remark

Some care is necessary here. We have chosen our terminology in such a way that the empty set is a heap, the empty subset is a subheap of every heap, but normal subheaps are non-empty by definition. As a consequence:

(a) Subheaps of a heap form a complete lattice (every intersection of subheaps is a subheap).

(b) Congruences on a heap form a complete lattice (every intersection of congruences is a congruence).

(c) Normal heaps of a heap do not form a lattice in general, but only a partially ordered set, because the intersection of two normal subheaps can be empty.

A *congruence* on a heap $(X, [-,-,-])$ is an equivalence relation $\sim$ on the set X such that $[x,y,z] \sim [x',y',z']$, for every $x,x',y,y',z,z' \in X$ such that $x \sim x'$, $y \sim y'$ and $z \sim z'$.

For example. . .

In a commutative heap all non-empty subheaps are normal.

Congruences and ideals

For a good algebraic structure (a group $(G, +, -, 0)$ or a ring $(R, +, -, 0)$), there is a one-to-one correspondence $\{\,\text{congruences}\,\} \longleftrightarrow \{\,\text{equivalence classes of } 0\,\}$ (=normal subgroups of G, or ideals of R).

Congruences and ideals

For a good algebraic structure (a group $(G, +, -, 0)$ or a ring $(R, +, -, 0)$), there is a one-to-one correspondence $\{\text{congruences}\} \longleftrightarrow \{\text{equivalence classes of } 0\}$ (=normal subgroups of G, or ideals of R). In the case of heaps, there is not a zero, any element can be a zero, and therefore the situation becomes $\{\text{congruences}\} \longleftarrow \{\text{equivalence classes (of any element)}\}$

Congruences and ideals

There is an onto mapping $\{\,\text{normal subheaps}\,\} \to \{\,\text{congruences}\,\}$.

An example

Consider the heap $(\mathbb{Z}, [-,-,-])$ of integer numbers with $[a, b, c] = a - b + c$.

An example

Consider the heap $(\mathbb{Z}, [-,-,-])$ of integer numbers with $[a,b,c] = a - b + c$. The complete lattice of its subheaps is $\{\, a + b\mathbb{Z} \mid a, b \in \mathbb{Z} \,\} \cup \{\emptyset\}$.

An example

Consider the heap $(\mathbb{Z}, [-,-,-])$ of integer numbers with $[a,b,c] = a - b + c$. The complete lattice of its subheaps is $\{\, a + b\mathbb{Z} \mid a, b \in \mathbb{Z} \,\} \cup \{\emptyset\}$. The set of its normal subheaps is $\{\, a + b\mathbb{Z} \mid a, b \in \mathbb{Z} \,\}$.

An example

Consider the heap $(\mathbb{Z}, [-,-,-])$ of integer numbers with $[a,b,c] = a - b + c$. The complete lattice of its subheaps is $\{\, a + b\mathbb{Z} \mid a, b \in \mathbb{Z} \,\} \cup \{\emptyset\}$. The set of its normal subheaps is $\{\, a + b\mathbb{Z} \mid a, b \in \mathbb{Z} \,\}$. Its congruences are the congruences $\equiv_n$ modulo n, and the complete lattice of congruence is $\{\, \equiv_n \mid n \in \mathbb{N} \,\}$, which is isomorphic to the lattice $(\mathbb{N}, |)$ with 0 as its greatest element and 1 as its least element.

An example

There is an onto mapping $\{\text{ normal subheaps }\} \to \{\text{ congruences }\}$, $S \mapsto\ \sim_S$, where $x \sim_S y$ if $[x, y, s] \in S$ for every $s \in S$.

An example

There is an onto mapping $\{\text{normal subheaps}\} \to \{\text{congruences}\}$, $S \mapsto \sim_S$, where $x \sim_S y$ if $[x, y, s] \in S$ for every $s \in S$.

In our example $(\mathbb{Z}, [-,-,-])$, that onto mapping is the correspondence $a + b\mathbb{Z} \mapsto$ congruence $\equiv_{|b|}$ modulo $|b|$. This is an onto mapping, but is not a bijection. Of course, $a + b\mathbb{Z} = c + d\mathbb{Z}$ if and only if $|b| = |d|$ and $a \equiv_{|b|} c$. In the next proposition, we will see that in order to get a one-to-one correspondence, that is, a bijection, it suffices to fix an element $e \in \mathbb{Z}$, and associate with any normal subheap $e + b\mathbb{Z}$ containing e the congruence $\equiv_{|b|}$ modulo $|b|$.

The situation for a generic heap (X, p) is the following:

Proposition

[Brzeziński] *Let X be a heap and e be a fixed element of X. Then there is a lattice isomorphism between the lattice of all congruences on the heap X and the lattice of all normal subheaps of X that contain e. It associates with any congruence $\sim$ the equivalence class $[e]_\sim$ of e. Conversely, it associates with any normal subheap S of X with $e \in S$ the congruence $\sim_S$ on X defined, for every $x, y \in X$, by $x \sim_S y$ if there exists $s \in S$ such that $[x, y, s] \in S$.*

The situation for a generic heap (X, p) is the following:

Proposition

[Brzeziński] *Let X be a heap and e be a fixed element of X. Then there is a lattice isomorphism between the lattice of all congruences on the heap X and the lattice of all normal subheaps of X that contain e. It associates with any congruence $\sim$ the equivalence class $[e]_\sim$ of e. Conversely, it associates with any normal subheap S of X with $e \in S$ the congruence $\sim_S$ on X defined, for every $x, y \in X$, by $x \sim_S y$ if there exists $s \in S$ such that $[x, y, s] \in S$.*

For any two normal subheaps S, T of a heap X, we have that $\sim_S \ \subseteq \ \sim_T$ if and only if, for every $x, y \in X$ and every $s \in S$ such that $[x, y, s] \in S$, there exists $t \in T$ such that $[x, y, t] \in T$.

Congruences and normal subheaps

By the previous proposition the lattice of all congruences on a heap X is isomorphic to the lattice of all normal subgroups of any of the groups (X, b_x).

Congruences and normal subheaps

By the previous proposition the lattice of all congruences on a heap X is isomorphic to the lattice of all normal subgroups of any of the groups (X, b_x). In particular, the lattice of all congruences on a heap is a complete modular lattice.

Congruences and normal subheaps

By the previous proposition the lattice of all congruences on a heap X is isomorphic to the lattice of all normal subgroups of any of the groups (X, b_x). In particular, the lattice of all congruences on a heap is a complete modular lattice.

Theorem

Let X be a heap. On the set $\mathcal{N}(X)$ of all normal subheaps of X define a pre-order $\preceq$ setting, for all $M, N \in \mathcal{N}(X)$, $M \preceq N$ if for every $x, y \in X$ and $s \in M$ such that $[x, y, s] \in M$ there exists $t \in N$ such that $[x, y, t] \in N$. Let $\simeq$ be the equivalence relation on $\mathcal{N}(X)$ associated to the pre-order $\preceq$. Then the partially ordered set $\mathcal{N}(X)/\simeq$ is order isomorphic to the partially ordered set $\mathcal{C}(X)$ of all congruences of the heap X.

Commutators of two congruences in a heap

Now let us consider the problem of determining a natural notion of commutator for a heap.

Commutators of two congruences in a heap

Now let us consider the problem of determining a natural notion of commutator for a heap. Let R and S be two congruences on a heap X, and let $R \times_X S$ be the set of all triples $(x, y, z) \in X^3$ such that xRy and ySz. Notice that $R \times_X S$ is a subheap of X^3. A canonical *connector* between R and S is the mapping

$$p \colon R \times_X S \to X$$

defined by $p(x, y, z) = [x, y, z]$ for every $(x, y, z) \in R \times_X S$, provided that $xS[x, y, z]$ and $[x, y, z]Rz$ for every $(x, y, z) \in R \times_X S$.

Commutators of two congruences in a heap

Now let us consider the problem of determining a natural notion of commutator for a heap. Let R and S be two congruences on a heap X, and let $R \times_X S$ be the set of all triples $(x, y, z) \in X^3$ such that xRy and ySz. Notice that $R \times_X S$ is a subheap of X^3. A canonical *connector* between R and S is the mapping

$$p\colon R \times_X S \to X$$

defined by $p(x, y, z) = [x, y, z]$ for every $(x, y, z) \in R \times_X S$, provided that $xS[x, y, z]$ and $[x, y, z]Rz$ for every $(x, y, z) \in R \times_X S$. The *commutator* of R and S is the smallest congruence $[R, S]$ on the heap X such that $R \times_X S \to X/[R, S]$, $(x, y, z) \mapsto [\,[x, y, z]\,]_{[R,S]}$, is a heap morphism.

Commutators of two congruences in a heap

Now let us consider the problem of determining a natural notion of commutator for a heap. Let R and S be two congruences on a heap X, and let $R \times_X S$ be the set of all triples $(x, y, z) \in X^3$ such that xRy and ySz. Notice that $R \times_X S$ is a subheap of X^3. A canonical *connector* between R and S is the mapping

$$p \colon R \times_X S \to X$$

defined by $p(x, y, z) = [x, y, z]$ for every $(x, y, z) \in R \times_X S$, provided that $xS[x, y, z]$ and $[x, y, z]Rz$ for every $(x, y, z) \in R \times_X S$. The *commutator* of R and S is the smallest congruence $[R, S]$ on the heap X such that $R \times_X S \to X/[R, S]$, $(x, y, z) \mapsto [\,[x, y, z]\,]_{[R,S]}$, is a heap morphism. That is, for every $x_1, y_1, z_1, x_2, y_2, z_2, x_3, y_3, z_3 \in X$ such that $x_i R y_i$ and $y_i S z_i$ for all $i = 1, 2, 3$, one has that

$$[[x_1, y_1, z_1], [x_2, y_2, z_2], [x_3, y_3, z_3]]\,[R, S]\,[[x_1, x_2, x_3], [y_1, y_2, y_3], [z_1, z_2, z_3]].$$

Commutators of two congruences in a heap

Let us compute the commutator of two congruences R, S on a heap (X, p).

Commutators of two congruences in a heap

Let us compute the commutator of two congruences R, S on a heap (X, p).

Theorem

Let R and S be two congruences on a heap (X, p). Fix an element e in X. Let $N := [e]_R$ and $M := [e]_S$ be the normal subgroups of the group (X, b_e) corresponding to the congruences R and S respectively. Then the commutator $[R, S]$ of R and S is the congruence on (X, p) corresponding to the normal subgroup $[N, M]$ of the group (X, b_e).

Commutators of two congruences in a heap

Let us compute the commutator of two congruences R, S on a heap (X, p).

Theorem

Let R and S be two congruences on a heap (X, p). Fix an element e in X. Let $N := [e]_R$ and $M := [e]_S$ be the normal subgroups of the group (X, b_e) corresponding to the congruences R and S respectively. Then the commutator $[R, S]$ of R and S is the congruence on (X, p) corresponding to the normal subgroup $[N, M]$ of the group (X, b_e).

In particular, a heap (X, p) is *abelian* if and only if $[X, X] = \{e\}$ in the group (X, b_e), that is, if and only if the group (X, b_e) is abelian. Since all the groups (X, b_y) are isomorphic, this is equivalent to all the groups (X, b_y) being abelian, that is, $[x, y, z] = [z, y, x]$ for every $x, y, z \in X$.

Idempotent endomorphisms and semidirect products of heaps

In any algebraic structure, idempotent endomorphisms are related to semidirect products.

Idempotent endomorphisms and semidirect products of heaps

In any algebraic structure, idempotent endomorphisms are related to semidirect products.

Proposition

Let $X \neq \emptyset$ be a heap, Y be a subheap of X, and ω a congruence on X. The following conditions are equivalent:
(a) *Y is a set of representatives of the equivalence classes of X modulo ω, that is, $Y \cap [x]_\omega$ is a singleton for every $x \in X$.*
(b) *There exists an idempotent heap endomorphism of X whose image is Y and whose kernel is ω.*
(c) *For every $e \in Y$, there exists an idempotent group endomorphism of the group (X, b_e) whose image is the subgroup Y of (X, b_e) and whose kernel is the normal subgroup $[e]_\omega$ of (X, b_e).*
(d) *The mapping $g \colon Y \to X/\omega$, defined by $g(y) = [y]_\omega$ for every $y \in Y$, is a heap isomorphism.*

Left near-trusses

A *left near-truss* $(X, [-,-,-], \cdot)$ is a set X endowed with a ternary operation $[-,-,-]$ and a binary operation $\cdot$, such that $(X, [-,-,-])$ is a heap, $(X, \cdot)$ is a semigroup, and *left distributivity* holds, that is,

$$x \cdot [y, z, w] = [x \cdot y, x \cdot z, x \cdot w]$$

for every $x, y, z, w \in X$.

Left near-trusses

A *left near-truss* $(X, [-,-,-], \cdot)$ is a set X endowed with a ternary operation $[-,-,-]$ and a binary operation $\cdot$, such that $(X, [-,-,-])$ is a heap, $(X, \cdot)$ is a semigroup, and *left distributivity* holds, that is,

$$x \cdot [y, z, w] = [x \cdot y, x \cdot z, x \cdot w]$$

for every $x, y, z, w \in X$. Similarly for *right near-trusses*, where left distributivity is replaced by *right distributivity*: $[y, z, w] \cdot x = [y \cdot x, z \cdot x, w \cdot x]$ for every $x, y, z, w \in X$.

Left near-trusses

A *left near-truss* $(X, [-,-,-], \cdot)$ is a set X endowed with a ternary operation $[-,-,-]$ and a binary operation $\cdot$, such that $(X, [-,-,-])$ is a heap, $(X, \cdot)$ is a semigroup, and *left distributivity* holds, that is,

$$x \cdot [y, z, w] = [x \cdot y, x \cdot z, x \cdot w]$$

for every $x, y, z, w \in X$. Similarly for *right near-trusses*, where left distributivity is replaced by *right distributivity*: $[y, z, w] \cdot x = [y \cdot x, z \cdot x, w \cdot x]$ for every $x, y, z, w \in X$. Clearly, the category of left near-trusses is isomorphic to the category of right near-trusses, it suffices to associate to any left near-truss $(X, [-,-,-], \cdot)$ its opposite right near-truss $(X, [-,-,-], \cdot^{\mathrm{op}})$.

Examples

(1) Let $(X, [-,-,-])$ be a heap and let

$$M(X) := \{ f \mid f\colon X \to X \}$$

be the set of all mappings from the set X to itself.

Examples

(1) Let $(X, [-,-,-])$ be a heap and let

$$M(X) := \{ f \mid f : X \to X \}$$

be the set of all mappings from the set X to itself. Define a ternary operation $[-,-,-]$ on $M(X)$ setting, for every $f, g, h \in M(X)$, $[f, g, h](x) = [f(x), g(x), h(x)]$ for all $x \in \ X$. Then $(M(X), [-,-,-])$ is also a heap (it is the direct product of $|X|$ copies of the heap $(X, [-,-,-])$).

Examples

(1) Let $(X, [-,-,-])$ be a heap and let

$$M(X) := \{ f \mid f \colon X \to X \}$$

be the set of all mappings from the set X to itself. Define a ternary operation $[-,-,-]$ on $M(X)$ setting, for every $f, g, h \in M(X)$, $[f, g, h](x) = [f(x), g(x), h(x)]$ for all $x \in \ X$. Then $(M(X), [-,-,-])$ is also a heap (it is the direct product of $|X|$ copies of the heap $(X, [-,-,-])$). Taking the composition of mappings as the binary operation $\cdot$, $M(X)$ becomes a *right* near-truss.

Examples

(1) Let $(X, [-,-,-])$ be a heap and let

$$M(X) := \{ f \mid f \colon X \to X \}$$

be the set of all mappings from the set X to itself. Define a ternary operation $[-,-,-]$ on $M(X)$ setting, for every $f, g, h \in M(X)$, $[f, g, h](x) = [f(x), g(x), h(x)]$ for all $x \in X$. Then $(M(X), [-,-,-])$ is also a heap (it is the direct product of $|X|$ copies of the heap $(X, [-,-,-])$). Taking the composition of mappings as the binary operation $\cdot$, $M(X)$ becomes a *right* near-truss.

(2) Let $(N, +, \cdot)$ be a left near-ring. Define a ternary operation $[-,-,-] \colon N \times N \times N \to N$ on N setting $[x, y, z] = x - y + z$ for every $x, y, z \in N$. Then $(N, [-,-,-], \cdot)$ is a left near-truss.

Examples

(3) Let $(B, *, \circ)$ be a left skew brace. Define a ternary operation $[-,-,-]\colon B \times B \times B \to B$ on B setting $[x, y, z] = x * (y^{-*}) * z$ for every $x, y, z \in B$. Then $(B, [-,-,-], \circ)$ is a left near-truss.

Examples

The right near-truss $M(X)$ of Example (1) is particularly interesting because:

Theorem

Every right near-truss is isomorphic to a subnear-truss of $M(X)$ for some heap X.

Lemma

Let $(X, [-,-,-], \cdot)$ *be a left near-truss and* y *be a fixed element of* X*.*

(a) *If* y *is a right zero for the semigroup* $(X, \cdot)$ *(that is,* $xy = y$ *for every* $x \in X$*), then* $(X, b_y, \cdot)$ *is a left near-ring.*

(b) *If* $(X, \cdot)$ *is a group and* y *is its identity, then* $(X, b_y, \cdot)$ *is a left skew brace.*

Trusses, endomorphism trusses

A *left truss* $(X, [-,-,-], \circ)$ is a left near-truss for which the heap $(X, [-,-,-])$ is abelian. Similarly, a *right truss* $(X, [-,-,-], \circ)$ is a right near-truss for which $(X, [-,-,-])$ is an abelian heap.

Trusses, endomorphism trusses

A *left truss* $(X, [-,-,-], \circ)$ is a left near-truss for which the heap $(X, [-,-,-])$ is abelian. Similarly, a *right truss* $(X, [-,-,-], \circ)$ is a right near-truss for which $(X, [-,-,-])$ is an abelian heap. A left truss that is also a right truss, is called a *truss*. Hence a truss $(X, [-,-,-], \circ)$ consists of an abelian heap $(X, [-,-,-])$, a semigroup $(X, \circ)$, and both distributivity laws hold.

The main example of ring with identity is, for any abelian group $(G, +)$, the endomorphism ring $(\mathrm{End}(G), +, \circ)$.

The main example of ring with identity is, for any abelian group $(G, +)$, the endomorphism ring $(\mathrm{End}(G), +, \circ)$. Similarly, the main example of truss is, for any abelian heap $(X, [-, -, -])$, the endomorphism truss $(\mathrm{End}(X), p, \circ)$ of $(X, [-, -, -])$. Here $\mathrm{End}(X)$ denotes the set of all heap endomorphisms of $(X, [-, -, -])$.

The main example of ring with identity is, for any abelian group $(G, +)$, the endomorphism ring $(\mathrm{End}(G), +, \circ)$. Similarly, the main example of truss is, for any abelian heap $(X, [-,-,-])$, the endomorphism truss $(\mathrm{End}(X), p, \circ)$ of $(X, [-,-,-])$. Here $\mathrm{End}(X)$ denotes the set of all heap endomorphisms of $(X, [-,-,-])$. The ternary operation p on $\mathrm{End}(X)$ is defined pointwise: for every $f, g, h \in \mathrm{End}(X)$, that is, for every $f, g, h \colon X \to X$ that are heap endomorphisms of X, we have that $p(f, g, h)(x) = [f(x), g(x), h(x)]$ for every $x \in X$.

Ideals in a left near-truss

A *congruence* on a left near-truss $(X, [-,-,-], \cdot)$ is an equivalence relation $\sim$ on the set X such that $[x, y, z] \sim [x', y', z']$ and $xy \sim x'y'$ for every $x, x', y, y', z, z' \in X$ such that $x \sim x'$, $y \sim y'$ and $z \sim z'$.

Ideals in a left near-truss

A *congruence* on a left near-truss $(X, [-,-,-], \cdot)$ is an equivalence relation $\sim$ on the set X such that $[x, y, z] \sim [x', y', z']$ and $xy \sim x'y'$ for every $x, x', y, y', z, z' \in X$ such that $x \sim x'$, $y \sim y'$ and $z \sim z'$. Congruences on a left near-truss form a complete lattice.

Ideals in a left near-truss

A *congruence* on a left near-truss $(X, [-,-,-], \cdot)$ is an equivalence relation $\sim$ on the set X such that $[x,y,z] \sim [x',y',z']$ and $xy \sim x'y'$ for every $x, x', y, y', z, z' \in X$ such that $x \sim x'$, $y \sim y'$ and $z \sim z'$. Congruences on a left near-truss form a complete lattice.

Lemma

Let $(X, [-,-,-], \cdot)$ be a left near-truss. For every normal subheap S of the heap $(X, [-,-,-])$, let $\sim_S$ be the corresponding congruence on the heap $(X, [-,-,-])$, defined, for every $x, y \in X$, by $x \sim_S y$ if there exists $s \in S$ such that $[x,y,s] \in S$. The following conditions are equivalent:

(a) *$\sim_S$ is a congruence for the left near-truss $(X, [-,-,-], \cdot)$.*

(b) *$[xp, xq, q] \in S$ and $[[p,q,x]y, xy, q] \in S$ for every $x, y \in X$ and every $p, q \in S$.*

Ideals in a left near-truss

An *ideal* in a left near-truss $(X, [-,-,-], \cdot)$ is any normal subheap S of $(X, [-,-,-])$ such that $[xp, xq, q] \in S$ and $[[p,q,x]y, xy, q] \in S$ for every $x, y \in X$ and every $p, q \in S$.

Ideals in a left near-truss

An *ideal* in a left near-truss $(X, [-,-,-], \cdot)$ is any normal subheap S of $(X, [-,-,-])$ such that $[xp, xq, q] \in S$ and $[[p, q, x]y, xy, q] \in S$ for every $x, y \in X$ and every $p, q \in S$.

Theorem

Let X be a left near-truss, $\mathcal{I}(X)$ the set of all ideals of X, and $\mathcal{C}(X)$ the set of all congruences of X. Then there is a mapping $\mathcal{I}(X) \to \mathcal{C}(X)$, $S \mapsto \sim_S$, which is a surjective mapping.

Theorem

Let $(X, [-,-,-], \cdot)$ *be a left near-truss, and fix an element* $y \in X$*. Then* $(X, b_y, \cdot)$ *is an algebra (in the sense of Universal Algebra) in which* (X, b_y) *is a group* $(X, *_y)$*,* $(X, \cdot)$ *is a semigroup, and* $w(x *_y z) = (wx) *_y (wy)^{-*} *_y (wz)$ *for every* $x, y, z, w \in X$*. Here* $(wy)^{-*}$ *denotes the inverse of the element* $w \cdot y$ *in the group* $(X, b_y) = (X, *_y)$*.*

Theorem

Let $(X, [-,-,-], \cdot)$ *be a left near-truss, and fix an element* $y \in X$*. Then* $(X, b_y, \cdot)$ *is an algebra (in the sense of Universal Algebra) in which* (X, b_y) *is a group* $(X, *_y)$*,* $(X, \cdot)$ *is a semigroup, and* $w(x *_y z) = (wx) *_y (wy)^{-*} *_y (wz)$ *for every* $x, y, z, w \in X$*. Here* $(wy)^{-*}$ *denotes the inverse of the element* $w \cdot y$ *in the group* $(X, b_y) = (X, *_y)$*.*

In view of this theorem, it is convenient to study the structures $(X, +, \cdot)$ for which $(X, +)$ is a group, not-necessarily abelian (so that probably we should be more careful and write also here $(X, +, 0, -)$ as one does correctly in Universal Algebra), $(X, \cdot)$ is a semigroup, and $w(x + z) = wx - (w \cdot 0) + wz$.

Theorem

Let $(X, [-,-,-], \cdot)$ *be a left near-truss, and fix an element* $y \in X$*. Then* $(X, b_y, \cdot)$ *is an algebra (in the sense of Universal Algebra) in which* (X, b_y) *is a group* $(X, *_y)$*,* $(X, \cdot)$ *is a semigroup, and* $w(x *_y z) = (wx) *_y (wy)^{-*} *_y (wz)$ *for every* $x, y, z, w \in X$*. Here* $(wy)^{-*}$ *denotes the inverse of the element* $w \cdot y$ *in the group* $(X, b_y) = (X, *_y)$*.*

In view of this theorem, it is convenient to study the structures $(X, +, \cdot)$ for which $(X, +)$ is a group, not-necessarily abelian (so that probably we should be more careful and write also here $(X, +, 0, -)$ as one does correctly in Universal Algebra), $(X, \cdot)$ is a semigroup, and $w(x + z) = wx - (w \cdot 0) + wz$. Let's call them J-rings (J for Jacobson), because our main example is, for any ring $(R, +, \cdot)$, the J-ring $(R, +, \circ)$, where $\circ$ is the *Jacobson multiplication* $x \circ y = x + y + xy$.

J-rings

Definition

A J-ring $(X, +, -, 0, \cdot)$ is a set X with two binary operations $+$ and $\cdot$, a unary operation $-$ and a 0-ary operation 0 satisfying:
(i) associativity of $+$;
(ii) $x + 0 = 0 + x = x$ for every $x \in X$;
(iii) $x + (-x) = (-x) + x = 0$ for every $x \in X$;
(iv) associativity of $\cdot$;
(v) "left weak distributivity" in the form
$z(x + y) = zx - (z \cdot 0) + zy$ for every $x, y, z \in X$.

Ideals in a J-ring

An *ideal* I in a J-ring $(X, +, \cdot)$ is a normal subgroup N of the group $(X, +)$ such that $xn - x \cdot 0 \in N$ and $(x + n)y - xy \in N$ for every $x, y \in X$ and every $n \in N$.

Ideals in a J-ring

An *ideal* I in a J-ring $(X, +, \cdot)$ is a normal subgroup N of the group $(X, +)$ such that $xn - x \cdot 0 \in N$ and $(x + n)y - xy \in N$ for every $x, y \in X$ and every $n \in N$.

Lemma

Let $(X, [-, -, -], \cdot)$ be a left near-truss and let e be an element of X. Then there is a lattice isomorphism between the lattice of all ideals of the J-ring $(X, b_e, \cdot)$ and the lattice of all congruences on $(X, [-, -, -], \cdot)$. This correspondence associates with every ideal N of the J-ring $(X, b_e, \cdot)$ the congruence $\sim_N$ on $(X, [-, -, -], \cdot)$ defined, for every $x, y \in X$, by $x \sim_N y$ if $x - y \in N$. Conversely, it associates to any congruence $\sim$ on $(X, [-, -, -], \cdot)$ the equivalence class $[e]_\sim$ of e modulo $\sim$.

Huq commutator and Smith commutator for left near-trusses, idempotent endomorphisms and semidirect product of left near-trusses, derivations of trusses, ...

The Baer-Kaplansky theorem

Theorem

[Baer 1943, Kaplansky1952)] *Two torsion abelian groups G and H are isomorphic if and only if their endomorphism rings* $\mathrm{End}(G)$ *and* $\mathrm{End}(H)$ *are isomorphic.*

The Baer-Kaplansky theorem

Theorem

[Baer 1943, Kaplansky1952)] *Two torsion abelian groups G and H are isomorphic if and only if their endomorphism rings $\mathrm{End}(G)$ and $\mathrm{End}(H)$ are isomorphic.*
Moreover, for every ring isomorphism $\Phi\colon \mathrm{End}(G) \to \mathrm{End}(H)$ there exists a unique group isomorphism $\varphi\colon G \to H$ such that $\Phi(\alpha) = \varphi\alpha\varphi^{-1}$ for every $\alpha \in \mathrm{End}(G)$.

The Baer-Kaplansky theorem

It is still unknown wither this is true for all abelian groups. More generally, it is still unknown when a right module M over an associative ring R is uniquely determined, up to isomorphism, by the ring $\mathrm{End}(M_R)$ of all its R-endomorphisms.

The Baer-Kaplansky theorem

Theorem

[Breaz and Brzeziński, 2022] *Two abelian groups G and H are isomorphic if and only if their endomorphism trusses* $\mathrm{End}_{\mathsf{Heap}}(G)$ *and* $\mathrm{End}_{\mathsf{Heap}}(H)$ *are isomorphic. Moreover, for every truss isomorphism* $\Phi\colon \mathrm{End}_{\mathsf{Heap}}(G) \to \mathrm{End}_{\mathsf{Heap}}(H)$*, there exists a unique heap isomorphism* $\varphi\colon G \to H$ *such that* $\Phi(\alpha) = \varphi\alpha\varphi^{-1}$ *for every* $\alpha \in \mathrm{End}_{\mathsf{Heap}}(G)$.