FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Lim-Inf Convergence[1]

Bartłomiej Skorulski
University of Białystok

Summary. This work continues the formalization of [7]. Theorems from Chapter III, Section 3, pp. 158–159 are proved.

MML Identifier: WAYBEL28.

The articles [5], [6], [10], [1], [15], [11], [17], [16], [12], [14], [8], [3], [4], [9], [2], and [13] provide the notation and terminology for this paper.

One can prove the following propositions:

(1) For every complete lattice L and for every net N in L holds $\inf N \leqslant \liminf N$.

(2) Let L be a complete lattice, N be a net in L, and x be an element of L. Suppose that for every subnet M of N holds $x = \liminf M$. Then $x = \liminf N$ and for every subnet M of N holds $x \geqslant \inf M$.

(3) Let L be a complete lattice, N be a net in L, and x be an element of L. Suppose $N \in \mathrm{NetUniv}(L)$. Suppose that for every subnet M of N such that $M \in \mathrm{NetUniv}(L)$ holds $x = \liminf M$. Then $x = \liminf N$ and for every subnet M of N such that $M \in \mathrm{NetUniv}(L)$ holds $x \geqslant \inf M$.

Let N be a non empty relational structure and let f be a map from N into N. We say that f is greater or equal to id if and only if:

(Def. 1) For every element j of the carrier of N holds $j \leqslant f(j)$.

We now state three propositions:

(4) For every reflexive non empty relational structure N holds id_N is greater or equal to id.

(5) Let N be a directed non empty relational structure and x, y be elements of N. Then there exists an element z of N such that $x \leqslant z$ and $y \leqslant z$.

(6) For every directed non empty relational structure N holds there exists a map from N into N which is greater or equal to id.

[1] This work has been supported by KBN Grant 8 T11C 018 12.

Let N be a directed non empty relational structure. One can verify that there exists a map from N into N which is greater or equal to id.

Let N be a reflexive non empty relational structure. Observe that there exists a map from N into N which is greater or equal to id.

Let L be a non empty 1-sorted structure, let N be a non empty net structure over L, and let f be a map from N into N. The functor $N \cdot f$ yielding a strict non empty net structure over L is defined by the conditions (Def. 2).

(Def. 2)(i) The relational structure of $N \cdot f =$ the relational structure of N, and

(ii) the mapping of $N \cdot f =$ (the mapping of N) $\cdot f$.

The following propositions are true:

(7) Let L be a non empty 1-sorted structure, N be a non empty net structure over L, and f be a map from N into N. Then the carrier of $N \cdot f =$ the carrier of N.

(8) Let L be a non empty 1-sorted structure, N be a non empty net structure over L, and f be a map from N into N. Then $N \cdot f = \langle$the carrier of N, the internal relation of N, (the mapping of N) $\cdot f\rangle$.

(9) Let L be a non empty 1-sorted structure, N be a transitive directed non empty relational structure, and f be a function from the carrier of N into the carrier of L. Then $\langle$the carrier of N, the internal relation of N, $f\rangle$ is a net in L.

Let L be a non empty 1-sorted structure, let N be a transitive directed non empty relational structure, and let f be a function from the carrier of N into the carrier of L. Note that $\langle$the carrier of N, the internal relation of N, $f\rangle$ is transitive directed and non empty.

We now state the proposition

(10) Let L be a non empty 1-sorted structure, N be a net in L, and p be a map from N into N. Then $N \cdot p$ is a net in L.

Let L be a non empty 1-sorted structure, let N be a net in L, and let p be a map from N into N. Note that $N \cdot p$ is transitive and directed.

Next we state two propositions:

(11) Let L be a non empty 1-sorted structure, N be a net in L, and p be a map from N into N. If $N \in \mathrm{NetUniv}(L)$, then $N \cdot p \in \mathrm{NetUniv}(L)$.

(12) Let L be a non empty 1-sorted structure and N, M be nets in L. Suppose the net structure of $N =$ the net structure of M. Then M is a subnet of N.

Let L be a non empty 1-sorted structure and let N be a net in L. Note that there exists a subnet of N which is strict.

The following proposition is true

(13) Let L be a non empty 1-sorted structure, N be a net in L, and p be a greater or equal to id map from N into N. Then $N \cdot p$ is a subnet of N.

Let L be a non empty 1-sorted structure, let N be a net in L, and let p be a greater or equal to id map from N into N. Then $N \cdot p$ is a strict subnet of N.

One can prove the following two propositions:

(14) Let L be a complete lattice, N be a net in L, and x be an element of L. Suppose $N \in \mathrm{NetUniv}(L)$. Suppose $x = \liminf N$ and for every subnet M of N such that $M \in \mathrm{NetUniv}(L)$ holds $x \geqslant \inf M$. Then $x = \liminf N$ and for every greater or equal to id map p from N into N holds $x \geqslant \inf(N \cdot p)$.

(15) Let L be a complete lattice, N be a net in L, and x be an element of L. Suppose $x = \liminf N$ and for every greater or equal to id map p from N into N holds $x \geqslant \inf(N \cdot p)$. Let M be a subnet of N. Then $x = \liminf M$.

Let L be a non empty relational structure. The lim inf convergence of L is a convergence class of L and is defined by the condition (Def. 3).

(Def. 3) Let N be a net in L. Suppose $N \in \mathrm{NetUniv}(L)$. Let x be an element of the carrier of L. Then $\langle N, x\rangle \in$ the lim inf convergence of L if and only if for every subnet M of N holds $x = \liminf M$.

We now state two propositions:

(16) Let L be a complete lattice, N be a net in L, and x be an element of L. Suppose $N \in \mathrm{NetUniv}(L)$. Then $\langle N, x\rangle \in$ the lim inf convergence of L if and only if for every subnet M of N such that $M \in \mathrm{NetUniv}(L)$ holds $x = \liminf M$.

(17) Let L be a non empty relational structure, N be a constant net in L, and M be a subnet of N. Then M is constant and the value of $N =$ the value of M.

Let L be a non empty relational structure. The functor $\xi(L)$ yielding a family of subsets of L is defined as follows:

(Def. 4) $\xi(L) =$ the topology of ConvergenceSpace(the lim inf convergence of L).

The following propositions are true:

(18) For every complete lattice L holds the lim inf convergence of L has (CONSTANTS) property.

(19) For every non empty relational structure L holds the lim inf convergence of L has (SUBNETS) property.

(20) For every continuous complete lattice L holds the lim inf convergence of L has (DIVERGENCE) property.

(21) Let L be a non empty relational structure and N, x be sets. If $\langle N, x\rangle \in$ the lim inf convergence of L, then $N \in \mathrm{NetUniv}(L)$.

(22) Let L be a non empty 1-sorted structure and C_1, C_2 be convergence classes of L. If $C_1 \subseteq C_2$, then the topology of ConvergenceSpace(C_2) $\subseteq$ the topology of ConvergenceSpace(C_1).

(23) Let L be a non empty reflexive relational structure. Then the lim inf convergence of $L \subseteq$ the Scott convergence of L.

(24) For all sets X, Y such that $X \subseteq Y$ holds $X \in$ the universe of Y.

(25) Let L be a non empty transitive reflexive relational structure and D be a directed non empty subset of L. Then $\mathrm{NetStr}(D) \in \mathrm{NetUniv}(L)$.

(26) For every complete lattice L and for every directed non empty subset D of L and for every subnet M of $\mathrm{NetStr}(D)$ holds $\liminf M = \sup D$.

(27) Let L be a non empty complete lattice and D be a directed non empty subset of L. Then $\langle \mathrm{NetStr}(D), \sup D \rangle \in$ the lim inf convergence of L.

(28) For every complete lattice L and for every subset U_1 of L such that $U_1 \in \xi(L)$ holds U_1 is property(S).

(29) For every non empty reflexive relational structure L and for every subset A of L such that $A \in \sigma(L)$ holds $A \in \xi(L)$.

(30) For every complete lattice L and for every subset A of L such that A is upper holds if $A \in \xi(L)$, then $A \in \sigma(L)$.

(31) Let L be a complete lattice and A be a subset of L. Suppose A is lower. Then $-A \in \xi(L)$ if and only if A is closed under directed sups.

References

[1] Grzegorz Bancerek. Complete lattices. *Formalized Mathematics*, 2(**5**):719–725, 1991.

[2] Grzegorz Bancerek. Bounds in posets and relational substructures. *Formalized Mathematics*, 6(**1**):81–91, 1997.

[3] Grzegorz Bancerek. Directed sets, nets, ideals, filters, and maps. *Formalized Mathematics*, 6(**1**):93–107, 1997.

[4] Grzegorz Bancerek. The "way-below" relation. *Formalized Mathematics*, 6(**1**):169–176, 1997.

[5] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[6] Czesław Byliński. Functions from a set to a set. *Formalized Mathematics*, 1(**1**):153–164, 1990.

[7] G. Gierz, K.H. Hofmann, K. Keimel, J.D. Lawson, M. Mislove, and D.S. Scott. *A Compendium of Continuous Lattices.* Springer-Verlag, Berlin, Heidelberg, New York, 1980.

[8] Adam Grabowski. Scott-continuous functions. *Formalized Mathematics*, 7(**1**):13–18, 1998.

[9] Artur Korniłowicz. On the topological properties of meet-continuous lattices. *Formalized Mathematics*, 6(**2**):269–277, 1997.

[10] Michał Muzalewski. Categories of groups. *Formalized Mathematics*, 2(**4**):563–571, 1991.

[11] Beata Padlewska and Agata Darmochwał. Topological spaces and continuous functions. *Formalized Mathematics*, 1(**1**):223–230, 1990.

[12] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[13] Andrzej Trybulec. Moore-Smith convergence. *Formalized Mathematics*, 6(**2**):213–225, 1997.

[14] Andrzej Trybulec. Scott topology. *Formalized Mathematics*, 6(**2**):311–319, 1997.

[15] Wojciech A. Trybulec. Partially ordered sets. *Formalized Mathematics*, 1(**2**):313–319, 1990.

[16] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[17] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

Received January 6, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

The Characterization of the Continuity of Topologies[1]

Grzegorz Bancerek
University of Białystok

Adam Naumowicz
University of Białystok

Summary. Formalization of [14, pp. 128–130], chapter II, section 4 (4.10, 4.11).

MML Identifier: WAYBEL29.

The terminology and notation used here are introduced in the following articles: [27], [23], [13], [10], [9], [21], [1], [30], [28], [24], [32], [22], [25], [31], [26], [12], [34], [29], [17], [15], [20], [6], [8], [3], [4], [33], [19], [7], [2], [16], [18], [5], and [11].

1. Preliminaries

The following propositions are true:

(1) Let S, T be non empty relational structures and f be a map from S into T. Suppose f is one-to-one and onto. Then $f \cdot f^{-1} = \mathrm{id}_T$ and $f^{-1} \cdot f = \mathrm{id}_S$ and f^{-1} is one-to-one and onto.

(2) Let X, Y be non empty sets, Z be a non empty relational structure, S be a non empty relational substructure of $Z^{[: X, Y :]}$, T be a non empty relational substructure of $(Z^Y)^X$, and f be a map from S into T. If f is currying, one-to-one, and onto, then f^{-1} is uncurrying.

(3) Let X, Y be non empty sets, Z be a non empty relational structure, S be a non empty relational substructure of $Z^{[: X, Y :]}$, T be a non empty relational substructure of $(Z^Y)^X$, and f be a map from T into S. If f is uncurrying, one-to-one, and onto, then f^{-1} is currying.

[1] This work has been supported by KBN Grant 8 T11C 018 12.

(4) Let X, Y be non empty sets, Z be a non empty poset, S be a non empty full relational substructure of $Z^{[\!: X,Y :\!]}$, T be a non empty full relational substructure of $(Z^Y)^X$, and f be a map from S into T. If f is currying, one-to-one, and onto, then f is isomorphic.

(5) Let X, Y be non empty sets, Z be a non empty poset, T be a non empty full relational substructure of $Z^{[\!: X,Y :\!]}$, S be a non empty full relational substructure of $(Z^Y)^X$, and f be a map from S into T. If f is uncurrying, one-to-one, and onto, then f is isomorphic.

(6) Let S_1, S_2, T_1, T_2 be relational structures. Suppose that

(i) the relational structure of $S_1 =$ the relational structure of S_2, and

(ii) the relational structure of $T_1 =$ the relational structure of T_2.

Let f be a map from S_1 into T_1. Suppose f is isomorphic. Let g be a map from S_2 into T_2. If $g = f$, then g is isomorphic.

(7) Let R, S, T be relational structures and f be a map from R into S. Suppose f is isomorphic. Let g be a map from S into T. Suppose g is isomorphic. Let h be a map from R into T. If $h = g \cdot f$, then h is isomorphic.

(8) Let T be an up-complete Scott non empty top-poset and S be a subset of T. Then S is closed if and only if S is directly closed and lower.

(9) Let S, T be up-complete Scott non empty top-posets and f be a map from S into T. If f is directed-sups-preserving, then f is continuous.

(10) Let X, Y, X_1, Y_1 be topological spaces. Suppose that

(i) the topological structure of $X =$ the topological structure of X_1, and

(ii) the topological structure of $Y =$ the topological structure of Y_1.

Then $[\!: X, Y :\!] = [\!: X_1, Y_1 :\!]$.

(11) Let X be a non empty topological space, L be a Scott up-complete non empty top-poset, and F be a non empty directed subset of $[X \to L]$. Then $\bigsqcup_{(L^{\text{the carrier of } X})} F$ is a continuous map from X into L.

(12) Let X be a non empty topological space and L be a Scott up-complete non empty top-poset. Then $[X \to L]$ is a directed-sups-inheriting relational substructure of $L^{\text{the carrier of } X}$.

(13) Let S_1, S_2 be topological structures. Suppose the topological structure of $S_1 =$ the topological structure of S_2. Let T_1, T_2 be non empty FR-structures. If the FR-structure of $T_1 =$ the FR-structure of T_2, then $[S_1 \to T_1] = [S_2 \to T_2]$.

One can check that every complete continuous top-lattice which is Scott is also injective and T_0.

One can check that there exists a top-lattice which is Scott, continuous, and complete.

Let X be a non empty topological space and let L be a Scott up-complete non empty top-poset. Note that $[X \to L]$ is up-complete.

One can prove the following propositions:

(14) Let I be a non empty set and J be a poset-yielding nonempty many sorted set indexed by I. Suppose that for every element i of I holds $J(i)$ is up-complete. Then $I\text{-prod}_{\mathrm{POS}}\, J$ is up-complete.

(15) Let I be a non empty set and J be a poset-yielding nonempty reflexive-yielding many sorted set indexed by I. Suppose that for every element i of I holds $J(i)$ is up-complete and lower-bounded. Let x, y be elements of $\prod J$. Then $x \ll y$ if and only if the following conditions are satisfied:

(i) for every element i of I holds $x(i) \ll y(i)$, and

(ii) there exists a finite subset K of I such that for every element i of I such that $i \notin K$ holds $x(i) = \bot_{J(i)}$.

Let X be a set and let L be a lower-bounded non empty reflexive antisymmetric relational structure. Observe that L^X is lower-bounded.

Let X be a non empty topological space and let L be a lower-bounded non empty top-poset. Note that $[X \to L]$ is lower-bounded.

Let L be an up-complete non empty poset. Note that every topological augmentation of L is up-complete and every topological augmentation of L which is Scott is also correct.

The following proposition is true

(16) Let S be an up-complete antisymmetric non empty reflexive relational structure and T be a non empty reflexive relational structure. Suppose the relational structure of S = the relational structure of T. Let A be a subset of S and C be a subset of T. If $A = C$ and A is inaccessible, then C is inaccessible.

Let L be an up-complete non empty poset. Observe that there exists a topological augmentation of L which is strict and Scott.

We now state two propositions:

(17) Let L be an up-complete non empty poset and S_1, S_2 be Scott topological augmentations of L. Then the topology of S_1 = the topology of S_2.

(18) Let S_1, S_2 be up-complete antisymmetric non empty reflexive FR-structures. Suppose the FR-structure of S_1 = the FR-structure of S_2 and S_1 is Scott. Then S_2 is Scott.

Let L be an up-complete non empty poset.

(Def. 1) ΣL is a strict Scott topological augmentation of L.

We now state two propositions:

(19) For every Scott up-complete non empty top-poset S holds ΣS = the FR-structure of S.

(20) Let L_1, L_2 be up-complete non empty posets. Suppose the relational structure of L_1 = the relational structure of L_2. Then $\Sigma L_1 = \Sigma L_2$.

Let S, T be up-complete non empty posets and let f be a map from S into T. The functor Σf yielding a map from ΣS into ΣT is defined as follows:

(Def. 2) $\Sigma f = f$.

Let S, T be up-complete non empty posets and let f be a directed-sups-preserving map from S into T. Observe that Σf is continuous.

One can prove the following propositions:

(21) Let S, T be up-complete non empty posets and f be a map from S into T. Then f is isomorphic if and only if Σf is isomorphic.

(22) For every non empty topological space X and for every Scott complete top-lattice S holds $[X \to S] = [X \to S]$.

Let X, Y be non empty topological spaces. The functor $\Theta(X, Y)$ yielding a map from $\langle$the topology of $[\!: X, Y :\!]$, $\subseteq\rangle$ into $[X \to \Sigma\langle$the topology of Y, $\subseteq\rangle]$ is defined as follows:

(Def. 3) For every open subset W of $[\!: X, Y :\!]$ holds $(\Theta(X,Y))(W) = \Theta_{\text{the carrier of } X}(W)$.

2. Some Natural Isomorphisms

Let X be a non empty topological space. The functor $\alpha(X)$ yielding a map from $[X \to$ the Sierpiński space$]$ into $\langle$the topology of X, $\subseteq\rangle$ is defined as follows:

(Def. 4) For every continuous map g from X into the Sierpiński space holds $(\alpha(X))(g) = g^{-1}(\{1\})$.

One can prove the following proposition

(23) For every non empty topological space X and for every open subset V of X holds $(\alpha(X))^{-1}(V) = \chi_{V,\text{the carrier of } X}$.

Let X be a non empty topological space. Note that $\alpha(X)$ is isomorphic.

Let X be a non empty topological space. One can verify that $(\alpha(X))^{-1}$ is isomorphic.

Let S be an injective T_0-space. One can verify that ΩS is Scott.

Let X be a non empty topological space. One can check that $[X \to$ the Sierpiński space$]$ is complete.

Next we state the proposition

(24) Ω(the Sierpiński space) $= \Sigma 2^1_\subseteq$.

Let M be a non empty set and let S be an injective T_0-space. One can verify that M-prod$_{\text{TOP}}(M \longmapsto S)$ is injective.

The following two propositions are true:

(25) For every non empty set M and for every complete continuous lattice L holds $\Omega(M\text{-prod}_{\text{TOP}}(M \longmapsto \Sigma L)) = \Sigma M\text{-prod}_{\text{POS}}(M \longmapsto L)$.

(26) For every non empty set M and for every injective T_0-space T holds $\Omega(M\text{-prod}_{\rm TOP}(M \longmapsto T)) = \Sigma M\text{-prod}_{\rm POS}(M \longmapsto \Omega T)$.

Let M be a non empty set and let X, Y be non empty topological spaces. The functor commute(X, M, Y) yielding a map from $[X \to M\text{-prod}_{\rm TOP}(M \longmapsto Y)]$ into $([X \to Y])^M$ is defined by:

(Def. 5) For every continuous map f from X into $M\text{-prod}_{\rm TOP}(M \longmapsto Y)$ holds $(\text{commute}(X, M, Y))(f) = \text{commute}(f)$.

Let M be a non empty set and let X, Y be non empty topological spaces. Note that commute(X, M, Y) is one-to-one and onto.

Let M be a non empty set and let X be a non empty topological space. Note that commute$(X, M,$ the Sierpiński space$)$ is isomorphic.

Next we state the proposition

(27) Let X, Y be non empty topological spaces, S be a Scott topological augmentation of $\langle$the topology of Y, $\subseteq\rangle$, and f_1, f_2 be elements of $[X \to S]$. If $f_1 \leqslant f_2$, then $G_{f_1} \subseteq G_{f_2}$.

3. The Poset of Open Sets

The following propositions are true:

(28) Let Y be a T_0-space. Then the following statements are equivalent

(i) for every non empty topological space X and for every Scott continuous complete top-lattice L and for every Scott topological augmentation T of $[Y \to L]$ there exists a map f from $[X \to T]$ into $[[\!: X, Y :\!] \to L]$ and there exists a map g from $[[\!: X, Y :\!] \to L]$ into $[X \to T]$ such that f is uncurrying, one-to-one, and onto and g is currying, one-to-one, and onto,

(ii) for every non empty topological space X and for every Scott continuous complete top-lattice L and for every Scott topological augmentation T of $[Y \to L]$ there exists a map f from $[X \to T]$ into $[[\!: X, Y :\!] \to L]$ and there exists a map g from $[[\!: X, Y :\!] \to L]$ into $[X \to T]$ such that f is uncurrying and isomorphic and g is currying and isomorphic.

(29) Let Y be a T_0-space. Then $\langle$the topology of Y, $\subseteq\rangle$ is continuous if and only if for every non empty topological space X holds $\Theta(X, Y)$ is isomorphic.

(30) Let Y be a T_0-space. Then $\langle$the topology of Y, $\subseteq\rangle$ is continuous if and only if for every non empty topological space X and for every continuous map f from X into $\Sigma\langle$the topology of Y, $\subseteq\rangle$ holds G_f is an open subset of $[\!: X, Y :\!]$.

(31) Let Y be a T_0-space. Then $\langle$the topology of Y, $\subseteq\rangle$ is continuous if and only if $\{\langle W, y\rangle; W$ ranges over open subsets of Y, y ranges over elements of Y: $y \in W\}$ is an open subset of $[\!: \Sigma\langle$the topology of Y, $\subseteq\rangle, Y :\!]$.

(32) Let Y be a T_0-space. Then $\langle$the topology of Y, $\subseteq\rangle$ is continuous if and only if for every element y of Y and for every open neighbourhood V of y there exists an open subset H of $\Sigma\langle$the topology of Y, $\subseteq\rangle$ such that $V \in H$ and $\bigcap H$ is a neighbourhood of y.

4. The Poset of Scott Open Sets

One can prove the following propositions:

(33) Let R_1, R_2, R_3 be non empty relational structures and f_1 be a map from R_1 into R_3. Suppose f_1 is isomorphic. Let f_2 be a map from R_2 into R_3. Suppose $f_2 = f_1$ and f_2 is isomorphic. Then the relational structure of $R_1 =$ the relational structure of R_2.

(34) Let L be a complete lattice. Then $\langle\sigma(L), \subseteq\rangle$ is continuous if and only if for every complete lattice S holds $\sigma([\!:S,\, L:\!]) =$ the topology of $[\!:\Sigma S,\, \Sigma L:\!]$.

(35) Let L be a complete lattice. Then the following statements are equivalent

(i) for every complete lattice S holds $\sigma([\!:S,\, L:\!]) =$ the topology of $[\!:\Sigma S,\, \Sigma L:\!]$,

(ii) for every complete lattice S holds the topological structure of $\Sigma[\!:S,\, L:\!] = [\!:\Sigma S,\, \Sigma L:\!]$.

(36) Let L be a complete lattice. Then for every complete lattice S holds $\sigma([\!:S,\, L:\!]) =$ the topology of $[\!:\Sigma S,\, \Sigma L:\!]$ if and only if for every complete lattice S holds $\Sigma[\!:S,\, L:\!] = \Omega[\!:\Sigma S,\, \Sigma L:\!]$.

(37) Let L be a complete lattice. Then $\langle\sigma(L), \subseteq\rangle$ is continuous if and only if for every complete lattice S holds $\Sigma[\!:S,\, L:\!] = \Omega[\!:\Sigma S,\, \Sigma L:\!]$.

References

[1] Grzegorz Bancerek. Complete lattices. *Formalized Mathematics*, 2(**5**):719–725, 1991.

[2] Grzegorz Bancerek. Bounds in posets and relational substructures. *Formalized Mathematics*, 6(**1**):81–91, 1997.

[3] Grzegorz Bancerek. Directed sets, nets, ideals, filters, and maps. *Formalized Mathematics*, 6(**1**):93–107, 1997.

[4] Grzegorz Bancerek. The "way-below" relation. *Formalized Mathematics*, 6(**1**):169–176, 1997.

[5] Grzegorz Bancerek. Bases and refinements of topologies. *Formalized Mathematics*, 7(**1**):35–43, 1998.

[6] Grzegorz Bancerek. Continuous lattices of maps between T_0 spaces. *Formalized Mathematics*, 9(**1**):111–117, 2001.

[7] Grzegorz Bancerek. Retracts and inheritance. *Formalized Mathematics*, 9(**1**):77–85, 2001.

[8] Grzegorz Bancerek and Adam Naumowicz. Function spaces in the category of directed suprema preserving maps. *Formalized Mathematics*, 9(**1**):171–177, 2001.

[9] Czesław Byliński. Basic functions and operations on functions. *Formalized Mathematics*, 1(**1**):245–254, 1990.

[10] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[11] Czesław Byliński. Some basic properties of sets. *Formalized Mathematics*, 1(**1**):47–53, 1990.

[12] Agata Darmochwał. Families of subsets, subspaces and mappings in topological spaces. *Formalized Mathematics*, 1(**2**):257–261, 1990.

[13] Agata Darmochwał. Finite sets. *Formalized Mathematics*, 1(**1**):165–167, 1990.

[14] G. Gierz, K.H. Hofmann, K. Keimel, J.D. Lawson, M. Mislove, and D.S. Scott. *A Compendium of Continuous Lattices*. Springer-Verlag, Berlin, Heidelberg, New York, 1980.

[15] Adam Grabowski. Scott-continuous functions. Part II. *Formalized Mathematics*, 9(**1**):5–11, 2001.

[16] Adam Grabowski and Robert Milewski. Boolean posets, posets under inclusion and products of relational structures. *Formalized Mathematics*, 6(**1**):117–121, 1997.

[17] Jarosław Gryko. Injective spaces. *Formalized Mathematics*, 7(**1**):57–62, 1998.

[18] Artur Korniłowicz. Cartesian products of relations and relational structures. *Formalized Mathematics*, 6(**1**):145–152, 1997.

[19] Artur Korniłowicz. On the topological properties of meet-continuous lattices. *Formalized Mathematics*, 6(**2**):269–277, 1997.

[20] Artur Korniłowicz and Jarosław Gryko. Injective spaces. Part II. *Formalized Mathematics*, 9(**1**):41–47, 2001.

[21] Michał Muzalewski. Categories of groups. *Formalized Mathematics*, 2(**4**):563–571, 1991.

[22] Beata Padlewska. Families of sets. *Formalized Mathematics*, 1(**1**):147–152, 1990.

[23] Beata Padlewska. Locally connected spaces. *Formalized Mathematics*, 2(**1**):93–96, 1991.

[24] Beata Padlewska and Agata Darmochwał. Topological spaces and continuous functions. *Formalized Mathematics*, 1(**1**):223–230, 1990.

[25] Yozo Toda. The formalization of simple graphs. *Formalized Mathematics*, 5(**1**):137–144, 1996.

[26] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[27] Andrzej Trybulec. A Borsuk theorem on homotopy types. *Formalized Mathematics*, 2(**4**):535–545, 1991.

[28] Andrzej Trybulec. Many-sorted sets. *Formalized Mathematics*, 4(**1**):15–22, 1993.

[29] Andrzej Trybulec. Scott topology. *Formalized Mathematics*, 6(**2**):311–319, 1997.

[30] Wojciech A. Trybulec. Partially ordered sets. *Formalized Mathematics*, 1(**2**):313–319, 1990.

[31] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[32] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

[33] Mariusz Żynel. The equational characterization of continuous lattices. *Formalized Mathematics*, 6(**2**):199–205, 1997.

[34] Mariusz Żynel and Adam Guzowski. T_0 topological spaces. *Formalized Mathematics*, 5(**1**):75–77, 1996.

Received January 6, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Meet Continuous Lattices Revisited[1]

Artur Korniłowicz
University of Białystok

Summary. This work is a continuation of formalization of [10]. Theorems from Chapter III, Section 2, pp. 153–156 are proved.

MML Identifier: WAYBEL30.

The articles [25], [20], [8], [9], [1], [23], [18], [24], [19], [26], [22], [6], [3], [7], [14], [4], [17], [15], [16], [2], [11], [12], [13], [21], and [5] provide the terminology and notation for this paper.

The following two propositions are true:

(1) For every set x and for every non empty set D holds $x \cap \bigcup D = \bigcup \{x \cap d : d$ ranges over elements of $D\}$.

(2) Let R be a non empty reflexive transitive relational structure and D be a non empty directed subset of $\langle \mathrm{Ids}(R), \subseteq \rangle$. Then $\bigcup D$ is an ideal of R.

Let R be a non empty reflexive transitive relational structure. Observe that $\langle \mathrm{Ids}(R), \subseteq \rangle$ is up-complete.

We now state two propositions:

(3) Let R be a non empty reflexive transitive relational structure and D be a non empty directed subset of $\langle \mathrm{Ids}(R), \subseteq \rangle$. Then $\sup D = \bigcup D$.

(4) Let R be a semilattice, D be a non empty directed subset of $\langle \mathrm{Ids}(R), \subseteq \rangle$, and x be an element of $\langle \mathrm{Ids}(R), \subseteq \rangle$. Then $\sup(\{x\} \sqcap D) = \bigcup \{x \cap d : d$ ranges over elements of $D\}$.

Let R be a semilattice. Observe that $\langle \mathrm{Ids}(R), \subseteq \rangle$ satisfies MC.

Let R be a non empty trivial relational structure. Note that every topological augmentation of R is trivial.

Next we state three propositions:

[1] This work has been supported by KBN Grant 8 T11C 018 12.

(5) Let S be a Scott complete top-lattice, T be a complete lattice, and A be a Scott topological augmentation of T. Suppose the relational structure of $S =$ the relational structure of T. Then the FR-structure of $A =$ the FR-structure of S.

(6) Let N be a Lawson complete top-lattice, T be a complete lattice, and A be a Lawson correct topological augmentation of T. Suppose the relational structure of $N =$ the relational structure of T. Then the FR-structure of $A =$ the FR-structure of N.

(7) Let N be a Lawson complete top-lattice, S be a Scott topological augmentation of N, A be a subset of N, and J be a subset of S. If $A = J$ and J is closed, then A is closed.

Let A be a complete lattice. Observe that $\lambda(A)$ is non empty.

Let S be a Scott complete top-lattice. Observe that $\langle\sigma(S), \subseteq\rangle$ is complete and non trivial.

Let N be a Lawson complete top-lattice. Observe that $\langle\sigma(N), \subseteq\rangle$ is complete and non trivial and $\langle\lambda(N), \subseteq\rangle$ is complete and non trivial.

The following propositions are true:

(8) Let T be a non empty reflexive relational structure. Then $\sigma(T) \subseteq \{W \setminus {\uparrow}F; W$ ranges over subsets of T, F ranges over subsets of T: $W \in \sigma(T) \ \wedge$ F is finite$\}$.

(9) For every Lawson complete top-lattice N holds $\lambda(N) =$ the topology of N.

(10) For every Lawson complete top-lattice N holds $\sigma(N) \subseteq \lambda(N)$.

(11) Let M, N be complete lattices. Suppose the relational structure of $M =$ the relational structure of N. Then $\lambda(M) = \lambda(N)$.

(12) For every Lawson complete top-lattice N and for every subset X of N holds $X \in \lambda(N)$ iff X is open.

Let us note that every reflexive non empty FR-structure which is trivial and topological space-like is also Scott.

Let us observe that every complete top-lattice which is trivial is also Lawson.

Let us note that there exists a complete top-lattice which is strict, continuous, lower-bounded, meet-continuous, and Scott.

One can verify that there exists a complete top-lattice which is strict, continuous, compact, Hausdorff, and Lawson.

Next we state the proposition

(13) Let N be a meet-continuous lattice and A be a subset of N. If A has the property (S), then ${\uparrow}A$ has the property (S).

Let N be a meet-continuous lattice and let A be a property(S) subset of N. Note that ${\uparrow}A$ is property(S).

We now state several propositions:

(14) Let N be a meet-continuous Lawson complete top-lattice, S be a Scott topological augmentation of N, and A be a subset of N. If $A \in \lambda(N)$, then $\uparrow A \in \sigma(S)$.

(15) Let N be a meet-continuous Lawson complete top-lattice, S be a Scott topological augmentation of N, A be a subset of N, and J be a subset of S. If $A = J$, then if A is open, then $\uparrow J$ is open.

(16) Let N be a meet-continuous Lawson complete top-lattice, S be a Scott topological augmentation of N, x be a point of S, y be a point of N, and J be a basis of y. If $x = y$, then $\{\uparrow A; A$ ranges over subsets of N: $A \in J\}$ is a basis of x.

(17) Let N be a meet-continuous Lawson complete top-lattice, S be a Scott topological augmentation of N, X be an upper subset of N, and Y be a subset of S. If $X = Y$, then $\operatorname{Int} X = \operatorname{Int} Y$.

(18) Let N be a meet-continuous Lawson complete top-lattice, S be a Scott topological augmentation of N, X be a lower subset of N, and Y be a subset of S. If $X = Y$, then $\overline{X} = \overline{Y}$.

(19) Let M, N be complete lattices, L_1 be a Lawson correct topological augmentation of M, and L_2 be a Lawson correct topological augmentation of N. Suppose $\langle \sigma(N), \subseteq \rangle$ is continuous. Then the topology of $[\!: L_1, (L_2$ **qua** topological space$) :\!] = \lambda([\!: M, N :\!])$.

(20) Let M, N be complete lattices, P be a Lawson correct topological augmentation of $[\!: M, N :\!]$, Q be a Lawson correct topological augmentation of M, and R be a Lawson correct topological augmentation of N. Suppose $\langle \sigma(N), \subseteq \rangle$ is continuous. Then the topological structure of $P = [\!: Q$, $(R$ **qua** topological space$) :\!]$.

(21) For every meet-continuous Lawson complete top-lattice N and for every element x of N holds $x \sqcap \square$ is continuous.

Let N be a meet-continuous Lawson complete top-lattice and let x be an element of N. Observe that $x \sqcap \square$ is continuous.

One can prove the following propositions:

(22) For every meet-continuous Lawson complete top-lattice N such that $\langle \sigma(N), \subseteq \rangle$ is continuous holds N satisfies conditions of topological semilattice.

(23) Let N be a meet-continuous Lawson complete top-lattice. Suppose $\langle \sigma(N), \subseteq \rangle$ is continuous. Then N is Hausdorff if and only if for every subset X of $[\!: N$, $(N$ **qua** topological space$) :\!]$ such that $X =$ the internal relation of N holds X is closed.

Let N be a non empty reflexive relational structure and let X be a subset of the carrier of N. The functor X^0 yields a subset of N and is defined by:

(Def. 1) $X^0 = \{u; u$ ranges over elements of N: $\bigwedge_{D\colon \text{non empty directed subset of } N} (u \leqslant$

$\sup D \ \Rightarrow\ X \cap D \neq \emptyset)\}$.

Let N be a non empty reflexive antisymmetric relational structure and let X be an empty subset of the carrier of N. One can check that X^0 is empty.

One can prove the following propositions:

(24) For every non empty reflexive relational structure N and for all subsets A, J of N such that $A \subseteq J$ holds $A^0 \subseteq J^0$.

(25) For every non empty reflexive relational structure N and for every element x of N holds $\uparrow x^0 = \twoheaduparrow x$.

(26) For every Scott top-lattice N and for every upper subset X of N holds $\mathrm{Int}\, X \subseteq X^0$.

(27) For every non empty reflexive relational structure N and for all subsets X, Y of N holds $X^0 \cup Y^0 \subseteq X \cup Y^0$.

(28) For every meet-continuous lattice N and for all upper subsets X, Y of N holds $X^0 \cup Y^0 = X \cup Y^0$.

(29) Let S be a meet-continuous Scott top-lattice and F be a finite subset of S. Then $\mathrm{Int} {\uparrow} F \subseteq \bigcup\{\twoheaduparrow x; x$ ranges over elements of S: $x \in F\}$.

(30) Let N be a Lawson complete top-lattice. Then N is continuous if and only if N is meet-continuous and Hausdorff.

Let us note that every complete top-lattice which is continuous and Lawson is also Hausdorff and every complete top-lattice which is meet-continuous, Lawson, and Hausdorff is also continuous.

Let N be a non empty FR-structure. We say that N has small semilattices if and only if the condition (Def. 2) is satisfied.

(Def. 2) Let x be a point of N. Then there exists a generalized basis J of x such that for every subset A of N if $A \in J$, then $\mathrm{sub}(A)$ is meet-inheriting.

We say that N has compact semilattices if and only if the condition (Def. 3) is satisfied.

(Def. 3) Let x be a point of N. Then there exists a generalized basis J of x such that for every subset A of N if $A \in J$, then $\mathrm{sub}(A)$ is meet-inheriting and A is compact.

We say that N has open semilattices if and only if the condition (Def. 4) is satisfied.

(Def. 4) Let x be a point of N. Then there exists a basis J of x such that for every subset A of N if $A \in J$, then $\mathrm{sub}(A)$ is meet-inheriting.

One can verify the following observations:

- $*$ every non empty topological space-like FR-structure which has open semilattices has also small semilattices,
- $*$ every non empty topological space-like FR-structure which has compact semilattices has also small semilattices,

- $*$ every non empty FR-structure which is anti-discrete has small semilattices and open semilattices, and
- $*$ every non empty FR-structure which is reflexive, trivial, and topological space-like has compact semilattices.

Let us mention that there exists a top-lattice which is strict, trivial, and lower.

We now state several propositions:

(31) Let N be top-poset with g.l.b.'s satisfying conditions of topological semilattice and C be a subset of N. If $\mathrm{sub}(C)$ is meet-inheriting, then $\mathrm{sub}(\overline{C})$ is meet-inheriting.

(32) Let N be a meet-continuous Lawson complete top-lattice and S be a Scott topological augmentation of N. Then for every point x of S there exists a basis J of x such that for every subset W of S such that $W \in J$ holds W is a filter of S if and only if N has open semilattices.

(33) Let N be a Lawson complete top-lattice, S be a Scott topological augmentation of N, and x be an element of N. Then $\{\inf A; A$ ranges over subsets of S: $x \in A \ \wedge \ A \in \sigma(S)\} \subseteq \{\inf J; J$ ranges over subsets of N: $x \in J \ \wedge \ J \in \lambda(N)\}$.

(34) Let N be a meet-continuous Lawson complete top-lattice, S be a Scott topological augmentation of N, and x be an element of N. Then $\{\inf A; A$ ranges over subsets of S: $x \in A \ \wedge \ A \in \sigma(S)\} = \{\inf J; J$ ranges over subsets of N: $x \in J \ \wedge \ J \in \lambda(N)\}$.

(35) Let N be a meet-continuous Lawson complete top-lattice. Then N is continuous if and only if N has open semilattices and $\langle \sigma(N), \subseteq \rangle$ is continuous.

One can check that every Lawson complete top-lattice which is continuous has open semilattices.

Let N be a continuous Lawson complete top-lattice. One can check that $\langle \sigma(N), \subseteq \rangle$ is continuous.

We now state several propositions:

(36) Every continuous Lawson complete top-lattice is compact and Hausdorff and has open semilattices and satisfies conditions of topological semilattice.

(37) Every Hausdorff Lawson complete top-lattice with open semilattices satisfying conditions of topological semilattice has compact semilattices.

(38) Let N be a meet-continuous Hausdorff Lawson complete top-lattice and x be an element of N. Then $x = \bigsqcup_N \{\inf V; V$ ranges over subsets of N: $x \in V \ \wedge \ V \in \lambda(N)\}$.

(39) Let N be a meet-continuous Lawson complete top-lattice. Then N is continuous if and only if for every element x of N holds $x = \bigsqcup_N \{\inf V; V$

ranges over subsets of N: $x \in V \ \wedge \ V \in \lambda(N)\}$.

(40) Let N be a meet-continuous Lawson complete top-lattice. Then N is algebraic if and only if N has open semilattices and $\langle \sigma(N), \subseteq \rangle$ is algebraic.

Let N be a meet-continuous algebraic Lawson complete top-lattice. Note that $\langle \sigma(N), \subseteq \rangle$ is algebraic.

References

[1] Grzegorz Bancerek. Complete lattices. *Formalized Mathematics*, 2(**5**):719–725, 1991.

[2] Grzegorz Bancerek. Bounds in posets and relational substructures. *Formalized Mathematics*, 6(**1**):81–91, 1997.

[3] Grzegorz Bancerek. Directed sets, nets, ideals, filters, and maps. *Formalized Mathematics*, 6(**1**):93–107, 1997.

[4] Grzegorz Bancerek. The "way-below" relation. *Formalized Mathematics*, 6(**1**):169–176, 1997.

[5] Grzegorz Bancerek. Bases and refinements of topologies. *Formalized Mathematics*, 7(**1**):35–43, 1998.

[6] Grzegorz Bancerek. The Lawson topology. *Formalized Mathematics*, 7(**2**):163–168, 1998.

[7] Czesław Byliński. Galois connections. *Formalized Mathematics*, 6(**1**):131–143, 1997.

[8] Agata Darmochwał. Compact spaces. *Formalized Mathematics*, 1(**2**):383–386, 1990.

[9] Agata Darmochwał. Finite sets. *Formalized Mathematics*, 1(**1**):165–167, 1990.

[10] G. Gierz, K.H. Hofmann, K. Keimel, J.D. Lawson, M. Mislove, and D.S. Scott. *A Compendium of Continuous Lattices*. Springer-Verlag, Berlin, Heidelberg, New York, 1980.

[11] Adam Grabowski and Robert Milewski. Boolean posets, posets under inclusion and products of relational structures. *Formalized Mathematics*, 6(**1**):117–121, 1997.

[12] Artur Korniłowicz. Cartesian products of relations and relational structures. *Formalized Mathematics*, 6(**1**):145–152, 1997.

[13] Artur Korniłowicz. Definitions and properties of the join and meet of subsets. *Formalized Mathematics*, 6(**1**):153–158, 1997.

[14] Artur Korniłowicz. Meet–continuous lattices. *Formalized Mathematics*, 6(**1**):159–167, 1997.

[15] Artur Korniłowicz. On the topological properties of meet-continuous lattices. *Formalized Mathematics*, 6(**2**):269–277, 1997.

[16] Artur Korniłowicz. Introduction to meet-continuous topological lattices. *Formalized Mathematics*, 7(**2**):279–283, 1998.

[17] Robert Milewski. Algebraic lattices. *Formalized Mathematics*, 6(**2**):249–254, 1997.

[18] Beata Padlewska and Agata Darmochwał. Topological spaces and continuous functions. *Formalized Mathematics*, 1(**1**):223–230, 1990.

[19] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[20] Andrzej Trybulec. A Borsuk theorem on homotopy types. *Formalized Mathematics*, 2(**4**):535–545, 1991.

[21] Andrzej Trybulec. Baire spaces, Sober spaces. *Formalized Mathematics*, 6(**2**):289–294, 1997.

[22] Andrzej Trybulec. Scott topology. *Formalized Mathematics*, 6(**2**):311–319, 1997.

[23] Wojciech A. Trybulec. Partially ordered sets. *Formalized Mathematics*, 1(**2**):313–319, 1990.

[24] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[25] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

[26] Mirosław Wysocki and Agata Darmochwał. Subsets of topological spaces. *Formalized Mathematics*, 1(**1**):231–237, 1990.

Received January 6, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Weights of Continuous Lattices[1]

Robert Milewski
University of Białystok

Summary. This work is a continuation of formalization of [13]. Theorems from Chapter III, Section 4, pp. 170–171 are proved.

MML Identifier: `WAYBEL31`.

The papers [25], [20], [1], [9], [12], [10], [22], [3], [15], [2], [23], [19], [26], [24], [27], [21], [8], [18], [5], [11], [6], [17], [16], [4], [14], and [7] provide the terminology and notation for this paper.

In this article we present several logical schemes. The scheme *UparrowUnion* deals with a relational structure $\mathcal{A}$ and a unary predicate $\mathcal{P}$, and states that:

> Let S be a family of subsets of the carrier of $\mathcal{A}$. If $S = \{X; X$ ranges over subsets of $\mathcal{A} : \mathcal{P}[X]\}$, then $\uparrow\bigcup S = \bigcup\{\uparrow X; X$ ranges over subsets of $\mathcal{A} : \mathcal{P}[X]\}$

for all values of the parameters.

The scheme *DownarrowUnion* deals with a relational structure $\mathcal{A}$ and a unary predicate $\mathcal{P}$, and states that:

> Let S be a family of subsets of the carrier of $\mathcal{A}$. If $S = \{X; X$ ranges over subsets of $\mathcal{A} : \mathcal{P}[X]\}$, then $\downarrow\bigcup S = \bigcup\{\downarrow X; X$ ranges over subsets of $\mathcal{A} : \mathcal{P}[X]\}$

for all values of the parameters.

Let L_1 be a lower-bounded continuous sup-semilattice and let B_1 be a CLbasis of L_1 with bottom. One can verify that $\langle \mathrm{Ids}(\mathrm{sub}(B_1)), \subseteq \rangle$ is algebraic.

Let L_1 be a continuous sup-semilattice. The functor CLweight L_1 yields a cardinal number and is defined as follows:

(Def. 1) $\mathrm{CLweight}\, L_1 = \bigcap\{\overline{\overline{B_1}} : B_1$ ranges over CLbasis of L_1 with bottom$\}$.

We now state a number of propositions:

[1] This work has been supported by KBN Grant 8 T11C 018 12.

(1) For every topological structure T and for every basis b of T holds weight $T \subseteq \overline{\overline{b}}$.

(2) For every topological structure T there exists a basis b of T such that $\overline{\overline{b}} =$ weight T.

(3) For every continuous sup-semilattice L_1 and for every CLbasis B_1 of L_1 with bottom holds CLweight $L_1 \subseteq \overline{\overline{B_1}}$.

(4) For every continuous sup-semilattice L_1 there exists a CLbasis B_1 of L_1 with bottom such that $\overline{\overline{B_1}} =$ CLweight L_1.

(5) For every algebraic lower-bounded lattice L_1 holds CLweight $L_1 = \overline{\overline{\text{the carrier of CompactSublatt}(L_1)}}$.

(6) Let T be a non empty topological space and L_1 be a continuous sup-semilattice. If $\langle$the topology of T, $\subseteq\rangle = L_1$, then every CLbasis of L_1 with bottom is a basis of T.

(7) Let T be a non empty topological space and L_1 be a continuous lower-bounded lattice. Suppose $\langle$the topology of T, $\subseteq\rangle = L_1$. Let B_1 be a basis of T and B_2 be a subset of L_1. If $B_1 = B_2$, then finsups(B_2) is a CLbasis of L_1 with bottom.

(8) Let T be a T_0 non empty topological space and L_1 be a continuous lower-bounded sup-semilattice. If $\langle$the topology of T, $\subseteq\rangle = L_1$, then if T is infinite, then weight $T =$ CLweight L_1.

(9) Let T be a T_0 non empty topological space and L_1 be a continuous sup-semilattice. Suppose $\langle$the topology of T, $\subseteq\rangle = L_1$. Then $\overline{\overline{\text{the carrier of } T}} \subseteq \overline{\overline{\text{the carrier of } L_1}}$.

(10) For every T_0 non empty topological space T such that T is finite holds weight $T = \overline{\overline{\text{the carrier of } T}}$.

(11) Let T be a topological structure and L_1 be a continuous lower-bounded lattice. Suppose $\langle$the topology of T, $\subseteq\rangle = L_1$ and T is finite. Then CLweight $L_1 = \overline{\overline{\text{the carrier of } L_1}}$.

(12) Let L_1 be a continuous lower-bounded sup-semilattice, T_1 be a Scott topological augmentation of L_1, T_2 be a Lawson correct topological augmentation of L_1, and B_2 be a basis of T_2. Then $\{\uparrow V; V$ ranges over subsets of T_2: $V \in B_2\}$ is a basis of T_1.

(13) For all finite sets X, Y such that $X \subseteq Y$ and $\overline{\overline{X}} = \overline{\overline{Y}}$ holds $X = Y$.

(14) For every up-complete non empty poset L_1 such that L_1 is finite and for every element x of L_1 holds $x \in$ compactbelow(x).

(15) Every finite lattice is arithmetic.

One can check that every lattice which is finite is also arithmetic.

One can verify that there exists a relational structure which is trivial, reflexive, transitive, antisymmetric, lower-bounded, non empty, finite, and strict and has l.u.b.'s and g.l.b.'s.

One can prove the following proposition

(16) Let L_1 be a finite lattice and B_1 be a CLbasis of L_1 with bottom. Then $\overline{\overline{B_1}} = \mathrm{CLweight}\, L_1$ if and only if $B_1 =$ the carrier of CompactSublatt(L_1).

Let L_1 be a non empty reflexive relational structure, let A be a subset of the carrier of L_1, and let a be an element of L_1. The functor Way_Up(a, A) yields a subset of L_1 and is defined as follows:

(Def. 2) Way_Up$(a, A) = \text{↟}a \setminus {\uparrow}A$.

Next we state a number of propositions:

(17) For every non empty reflexive relational structure L_1 and for every element a of L_1 holds Way_Up$(a, \emptyset_{(L_1)}) = \text{↟}a$.

(18) For every non empty poset L_1 and for every subset A of L_1 and for every element a of L_1 such that $a \in {\uparrow}A$ holds Way_Up$(a, A) = \emptyset$.

(19) For every non empty finite reflexive transitive relational structure L_1 holds Ids(L_1) is finite.

(20) For every continuous lower-bounded sup-semilattice L_1 such that L_1 is infinite holds every CLbasis of L_1 with bottom is infinite.

(21) For every set d and for every finite sequence p and for every natural number i such that $i \in \operatorname{dom} p$ holds $(\langle d \rangle \frown p)(i+1) = p(i)$.

(22) For every finite sequence p and for every set x holds $(\langle x \rangle \frown p)_{\restriction 1} = p$.

(23) For every complete non empty poset L_1 and for every element x of L_1 such that x is compact holds $x = \inf \text{↟}x$.

(24) Let L_1 be a continuous lower-bounded sup-semilattice. Suppose L_1 is infinite. Let B_1 be a CLbasis of L_1 with bottom. Then $\overline{\overline{\{\mathrm{Way\_Up}(a, A); a \text{ ranges over elements of } L_1,\ A \text{ ranges over finite subsets of } L_1\colon a \in B_1 \land A \subseteq B_1\}}} \subseteq \overline{\overline{B_1}}$.

(25) For every Lawson complete top-lattice T and for every finite subset X of T holds $-{\uparrow}X$ is open and $-{\downarrow}X$ is open.

(26) Let L_1 be a continuous lower-bounded sup-semilattice, T be a Lawson correct topological augmentation of L_1, and B_1 be a CLbasis of L_1 with bottom. Then {Way_Up(a, A); a ranges over elements of L_1, A ranges over finite subsets of L_1: $a \in B_1 \land A \subseteq B_1$} is a basis of T.

(27) Let L_1 be a continuous lower-bounded sup-semilattice, T be a Scott topological augmentation of L_1, and b be a basis of T. Then $\{\text{↟}\inf u; u$ ranges over subsets of T: $u \in b\}$ is a basis of T.

(28) Let L_1 be a continuous lower-bounded sup-semilattice, T be a Scott topological augmentation of L_1, and B_1 be a basis of T. If B_1 is infinite, then $\{\inf u; u$ ranges over subsets of T: $u \in B_1\}$ is infinite.

(29) Let L_1 be a continuous lower-bounded sup-semilattice and T be a Scott topological augmentation of L_1. Then $\mathrm{CLweight}\, L_1 = \mathrm{weight}\, T$.

(30) Let L_1 be a continuous lower-bounded sup-semilattice and T be a Lawson correct topological augmentation of L_1. Then $\mathrm{CLweight}\, L_1 = \mathrm{weight}\, T$.

(31) Let L_1, L_2 be non empty relational structures. Suppose L_1 and L_2 are isomorphic. Then $\overline{\overline{\text{the carrier of } L_1}} = \overline{\overline{\text{the carrier of } L_2}}$.

(32) Let L_1 be a continuous lower-bounded sup-semilattice and B_1 be a CLbasis of L_1 with bottom. If $\overline{\overline{B_1}} = \mathrm{CLweight}\, L_1$, then $\mathrm{CLweight}\, L_1 = \mathrm{CLweight}\langle \mathrm{Ids}(\mathrm{sub}(B_1)), \subseteq \rangle$.

Let L_1 be a continuous lower-bounded sup-semilattice. Note that $\langle \sigma(L_1), \subseteq \rangle$ is continuous and has l.u.b.'s.

Next we state two propositions:

(33) For every continuous lower-bounded sup-semilattice L_1 holds $\mathrm{CLweight}\, L_1 \subseteq \mathrm{CLweight}\langle \sigma(L_1), \subseteq \rangle$.

(34) For every continuous lower-bounded sup-semilattice L_1 such that L_1 is infinite holds $\mathrm{CLweight}\, L_1 = \mathrm{CLweight}\langle \sigma(L_1), \subseteq \rangle$.

References

[1] Grzegorz Bancerek. Cardinal numbers. *Formalized Mathematics*, 1(**2**):377–382, 1990.

[2] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[3] Grzegorz Bancerek. Complete lattices. *Formalized Mathematics*, 2(**5**):719–725, 1991.

[4] Grzegorz Bancerek. Bounds in posets and relational substructures. *Formalized Mathematics*, 6(**1**):81–91, 1997.

[5] Grzegorz Bancerek. Directed sets, nets, ideals, filters, and maps. *Formalized Mathematics*, 6(**1**):93–107, 1997.

[6] Grzegorz Bancerek. The "way-below" relation. *Formalized Mathematics*, 6(**1**):169–176, 1997.

[7] Grzegorz Bancerek. Bases and refinements of topologies. *Formalized Mathematics*, 7(**1**):35–43, 1998.

[8] Grzegorz Bancerek. The Lawson topology. *Formalized Mathematics*, 7(**2**):163–168, 1998.

[9] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[10] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[11] Czesław Byliński. Galois connections. *Formalized Mathematics*, 6(**1**):131–143, 1997.

[12] Agata Darmochwał. Finite sets. *Formalized Mathematics*, 1(**1**):165–167, 1990.

[13] G. Gierz, K.H. Hofmann, K. Keimel, J.D. Lawson, M. Mislove, and D.S. Scott. *A Compendium of Continuous Lattices*. Springer-Verlag, Berlin, Heidelberg, New York, 1980.

[14] Adam Grabowski and Robert Milewski. Boolean posets, posets under inclusion and products of relational structures. *Formalized Mathematics*, 6(**1**):117–121, 1997.

[15] Katarzyna Jankowska. Transpose matrices and groups of permutations. *Formalized Mathematics*, 2(**5**):711–717, 1991.

[16] Artur Korniłowicz. On the topological properties of meet-continuous lattices. *Formalized Mathematics*, 6(**2**):269–277, 1997.

[17] Robert Milewski. Algebraic lattices. *Formalized Mathematics*, 6(**2**):249–254, 1997.

[18] Robert Milewski. Bases of continuous lattices. *Formalized Mathematics*, 7(**2**):285–294, 1998.

[19] Beata Padlewska and Agata Darmochwał. Topological spaces and continuous functions. *Formalized Mathematics*, 1(**1**):223–230, 1990.

[20] Alexander Yu. Shibakov and Andrzej Trybulec. The Cantor set. *Formalized Mathematics*, 5(**2**):233–236, 1996.

[21] Andrzej Trybulec. Scott topology. *Formalized Mathematics*, 6(**2**):311–319, 1997.

[22] Wojciech A. Trybulec. Groups. *Formalized Mathematics*, 1(**5**):821–827, 1990.

[23] Wojciech A. Trybulec. Partially ordered sets. *Formalized Mathematics*, 1(**2**):313–319, 1990.

[24] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[25] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

[26] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

[27] Mariusz Żynel and Adam Guzowski. T_0 topological spaces. *Formalized Mathematics*, 5(**1**):75–77, 1996.

Received January 6, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Representation Theorem for Finite Distributive Lattices

Marek Dudzicz
University of Białystok

Summary. In the article the representation theorem for finite distributive lattice as rings of sets is presented. Auxiliary concepts are introduced. Namely, the concept of the height of an element, the maximal element in a chain, immediate predecessor of an element and ring of sets. Besides the schemes of induction in finite lattice is proved.

MML Identifier: `LATTICE7`.

The notation and terminology used here are introduced in the following papers: [7], [1], [8], [6], [9], [3], [4], [2], and [5].

1. Induction in a Finite Lattice

Let L be a 1-sorted structure and let A, B be subsets of L. Let us observe that $A \subseteq B$ if and only if:

(Def. 1) For every element x of L such that $x \in A$ holds $x \in B$.

Let L be a lattice. Note that there exists a chain of L which is non empty.

Let L be a lattice and let x, y be elements of L. Let us assume that $x \leqslant y$. A non empty chain of L is called a chain of x, y if:

(Def. 2) $x \in$ it and $y \in$ it and for every element z of L such that $z \in$ it holds $x \leqslant z$ and $z \leqslant y$.

The following proposition is true

(1) For every lattice L and for all elements x, y of L such that $x \leqslant y$ holds $\{x, y\}$ is a chain of x, y.

ISSN 1426–2630

Let L be a finite lattice and let x be an element of L. The functor height x yields a natural number and is defined as follows:

(Def. 3) There exists a chain A of $\bot_L$, x such that height x = card A and for every chain A of $\bot_L$, x holds card $A \leqslant$ height x.

Next we state several propositions:

(2) For every finite lattice L and for all elements a, b of L such that $a < b$ holds height $a <$ height b.

(3) Let L be a finite lattice, C be a chain of L, and x, y be elements of L. If $x \in C$ and $y \in C$, then $x < y$ iff height $x <$ height y.

(4) Let L be a finite lattice, C be a chain of L, and x, y be elements of L. If $x \in C$ and $y \in C$, then $x = y$ iff height x = height y.

(5) Let L be a finite lattice, C be a chain of L, and x, y be elements of L. If $x \in C$ and $y \in C$, then $x \leqslant y$ iff height $x \leqslant$ height y.

(6) For every finite lattice L and for every element x of L holds height $x = 1$ iff $x = \bot_L$.

(7) For every non empty finite lattice L and for every element x of L holds height $x \geqslant 1$.

The scheme *LattInd* deals with a finite lattice $\mathcal{A}$ and a unary predicate $\mathcal{P}$, and states that:

For every element x of $\mathcal{A}$ holds $\mathcal{P}[x]$

provided the following requirement is met:

- For every element x of $\mathcal{A}$ such that for every element b of $\mathcal{A}$ such that $b < x$ holds $\mathcal{P}[b]$ holds $\mathcal{P}[x]$.

2. Join Irreducible Elements in a Finite Distributive Lattice

Let us mention that there exists a lattice which is distributive and finite.

Let L be a lattice and let x, y be elements of L. The predicate $x <_1 y$ is defined as follows:

(Def. 4) $x < y$ and it is not true that there exists an element z of L such that $x < z$ and $z < y$.

One can prove the following proposition

(8) Let L be a finite lattice and X be a non empty subset of L. Then there exists an element x of L such that $x \in X$ and for every element y of L such that $y \in X$ holds $x \not< y$.

Let L be a finite lattice and let A be a non empty chain of L. The functor $\max A$ yielding an element of L is defined by:

(Def. 5) For every element x of L such that $x \in A$ holds $x \leqslant \max A$ and $\max A \in A$.

The following proposition is true

(9) For every finite lattice L and for every element y of L such that $y \neq \bot_L$ there exists an element x of L such that $x <_1 y$.

Let L be a lattice. The functor Join-IRR L yielding a subset of L is defined by:

(Def. 6) Join-IRR L $= \{a; a$ ranges over elements of L: $a \neq \bot_L \wedge \bigwedge_{b,c:\text{element of } L} (a = b \sqcup c \Rightarrow a = b \vee a = c)\}$.

One can prove the following three propositions:

(10) Let L be a lattice and x be an element of L. Then $x \in$ Join-IRR L if and only if the following conditions are satisfied:

(i) $x \neq \bot_L$, and

(ii) for all elements b, c of L such that $x = b \sqcup c$ holds $x = b$ or $x = c$.

(11) Let L be a finite distributive lattice and x be an element of L. Suppose $x \in$ Join-IRR L. Then there exists an element z of L such that $z < x$ and for every element y of L such that $y < x$ holds $y \leqslant z$.

(12) For every distributive finite lattice L and for every element x of L holds $\sup(\downarrow x \cap \text{Join-IRR}\, L) = x$.

3. Representation Theorem

Let P be a relational structure. The functor LOWER P yields a non empty set and is defined as follows:

(Def. 7) LOWER $P = \{X; X$ ranges over subsets of P: X is lower$\}$.

The following two propositions are true:

(13) Let L be a distributive finite lattice. Then there exists a map r from L into $\langle \text{LOWER}\, \text{sub}(\text{Join-IRR}\, L), \subseteq \rangle$ such that r is isomorphic and for every element a of L holds $r(a) = \downarrow a \cap \text{Join-IRR}\, L$.

(14) For every distributive finite lattice L holds L and $\langle \text{LOWER}\, \text{sub}(\text{Join-IRR}\, L), \subseteq \rangle$ are isomorphic.

Ring of sets is defined by:

(Def. 8) It includes lattice of it.

Let us note that there exists a ring of sets which is non empty.

Let X be a non empty ring of sets. One can verify that $\langle X, \subseteq \rangle$ is distributive and has l.u.b.'s and g.l.b.'s.

One can prove the following propositions:

(15) For every finite lattice L holds LOWER sub(Join-IRR L) is a ring of sets.

(16) Let L be a finite lattice. Then L is distributive if and only if there exists a non empty ring of sets X such that L and $\langle X, \subseteq \rangle$ are isomorphic.

References

[1] Grzegorz Bancerek. Complete lattices. *Formalized Mathematics*, 2(**5**):719–725, 1991.

[2] Grzegorz Bancerek. Bounds in posets and relational substructures. *Formalized Mathematics*, 6(**1**):81–91, 1997.

[3] Grzegorz Bancerek. Directed sets, nets, ideals, filters, and maps. *Formalized Mathematics*, 6(**1**):93–107, 1997.

[4] Czesław Byliński. Galois connections. *Formalized Mathematics*, 6(**1**):131–143, 1997.

[5] Adam Grabowski and Robert Milewski. Boolean posets, posets under inclusion and products of relational structures. *Formalized Mathematics*, 6(**1**):117–121, 1997.

[6] Beata Padlewska and Agata Darmochwał. Topological spaces and continuous functions. *Formalized Mathematics*, 1(**1**):223–230, 1990.

[7] Wojciech A. Trybulec. Groups. *Formalized Mathematics*, 1(**5**):821–827, 1990.

[8] Wojciech A. Trybulec. Partially ordered sets. *Formalized Mathematics*, 1(**2**):313–319, 1990.

[9] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

Received January 6, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

The Field of Complex Numbers

Anna Justyna Milewska
University of Białystok

Summary. This article contains the definition and many facts about the field of complex numbers.

MML Identifier: `COMPLFLD`.

The articles [4], [1], [2], [5], [6], and [3] provide the terminology and notation for this paper.

The following propositions are true:

(1) $1_{\mathbb{C}} \neq 0_{\mathbb{C}}$.

(2) For all elements x_1, y_1, x_2, y_2 of $\mathbb{R}$ holds $(x_1 + y_1 i) + (x_2 + y_2 i) = (x_1 + x_2) + (y_1 + y_2)i$.

The strict double loop structure $\mathbb{C}_{\mathrm{F}}$ is defined by the conditions (Def. 1).

(Def. 1)(i) The carrier of $\mathbb{C}_{\mathrm{F}} = \mathbb{C}$,

(ii) the addition of $\mathbb{C}_{\mathrm{F}} = +_{\mathbb{C}}$,

(iii) the multiplication of $\mathbb{C}_{\mathrm{F}} = \cdot_{\mathbb{C}}$,

(iv) the unity of $\mathbb{C}_{\mathrm{F}} = 1_{\mathbb{C}}$, and

(v) the zero of $\mathbb{C}_{\mathrm{F}} = 0_{\mathbb{C}}$.

Let us observe that $\mathbb{C}_{\mathrm{F}}$ is non empty.

Let us observe that $\mathbb{C}_{\mathrm{F}}$ is add-associative right zeroed right complementable Abelian commutative associative left unital right unital distributive field-like and non degenerated.

We now state several propositions:

(3) For all elements x_1, y_1 of the carrier of $\mathbb{C}_{\mathrm{F}}$ and for all elements x_2, y_2 of $\mathbb{C}$ such that $x_1 = x_2$ and $y_1 = y_2$ holds $x_1 + y_1 = x_2 + y_2$.

(4) For every element x_1 of the carrier of $\mathbb{C}_{\mathrm{F}}$ and for every element x_2 of $\mathbb{C}$ such that $x_1 = x_2$ holds $-x_1 = -x_2$.

ISSN 1426–2630

(5) For all elements x_1, y_1 of the carrier of $\mathbb{C}_{\rm F}$ and for all elements x_2, y_2 of $\mathbb{C}$ such that $x_1 = x_2$ and $y_1 = y_2$ holds $x_1 - y_1 = x_2 - y_2$.

(6) For all elements x_1, y_1 of the carrier of $\mathbb{C}_{\rm F}$ and for all elements x_2, y_2 of $\mathbb{C}$ such that $x_1 = x_2$ and $y_1 = y_2$ holds $x_1 \cdot y_1 = x_2 \cdot y_2$.

(7) For every element x_1 of the carrier of $\mathbb{C}_{\rm F}$ and for every element x_2 of $\mathbb{C}$ such that $x_1 = x_2$ and $x_1 \neq 0_{\mathbb{C}_{\rm F}}$ holds ${x_1}^{-1} = {x_2}^{-1}$.

(8) Let x_1, y_1 be elements of the carrier of $\mathbb{C}_{\rm F}$ and x_2, y_2 be elements of $\mathbb{C}$. If $x_1 = x_2$ and $y_1 = y_2$ and $y_1 \neq 0_{\mathbb{C}_{\rm F}}$, then $\frac{x_1}{y_1} = \frac{x_2}{y_2}$.

(9) $0_{\mathbb{C}_{\rm F}} = 0_{\mathbb{C}}$.

(10) $\mathbf{1}_{\mathbb{C}_{\rm F}} = 1_{\mathbb{C}}$.

(11) $\mathbf{1}_{\mathbb{C}_{\rm F}} + \mathbf{1}_{\mathbb{C}_{\rm F}} \neq 0_{\mathbb{C}_{\rm F}}$.

Let z be an element of the carrier of $\mathbb{C}_{\rm F}$. The functor z^* yielding an element of $\mathbb{C}_{\rm F}$ is defined by:

(Def. 2) There exists an element z' of $\mathbb{C}$ such that $z = z'$ and $z^* = z'^*$.

Let z be an element of the carrier of $\mathbb{C}_{\rm F}$. The functor $|z|$ yielding an element of $\mathbb{R}$ is defined by:

(Def. 3) There exists an element z' of $\mathbb{C}$ such that $z = z'$ and $|z| = |z'|$.

We now state the proposition

(12) For every element x_1 of the carrier of $\mathbb{C}_{\rm F}$ and for every element x_2 of $\mathbb{C}$ such that $x_1 = x_2$ holds ${x_1}^* = {x_2}^*$.

In the sequel z, z_1, z_2, z_3, z_4 denote elements of the carrier of $\mathbb{C}_{\rm F}$.

One can prove the following propositions:

(13) $z_1 + (z_2 + z_3) = (z_1 + z_2) + z_3$.

(14) (The zero of $\mathbb{C}_{\rm F}$) $+ z = z$ and $z +$ the zero of $\mathbb{C}_{\rm F} = z$.

(15) $z_1 \cdot (z_2 \cdot z_3) = (z_1 \cdot z_2) \cdot z_3$.

(16) $z \cdot (z_1 + z_2) = z \cdot z_1 + z \cdot z_2$ and $(z_1 + z_2) \cdot z = z_1 \cdot z + z_2 \cdot z$.

(17) (The zero of $\mathbb{C}_{\rm F}$) $\cdot z =$ the zero of $\mathbb{C}_{\rm F}$ and $z \cdot$ the zero of $\mathbb{C}_{\rm F} =$ the zero of $\mathbb{C}_{\rm F}$.

(18) (The unity of $\mathbb{C}_{\rm F}$) $\cdot z = z$ and $z \cdot$ the unity of $\mathbb{C}_{\rm F} = z$.

(19) $-$the zero of $\mathbb{C}_{\rm F} =$ the zero of $\mathbb{C}_{\rm F}$.

(20) If $-z =$ the zero of $\mathbb{C}_{\rm F}$, then $z =$ the zero of $\mathbb{C}_{\rm F}$.

(21) $z + -z =$ the zero of $\mathbb{C}_{\rm F}$ and $-z + z =$ the zero of $\mathbb{C}_{\rm F}$.

(22) If $z_1 + z_2 =$ the zero of $\mathbb{C}_{\rm F}$, then $z_2 = -z_1$ and $z_1 = -z_2$.

(23) $--z = z$.

(24) If $-z_1 = -z_2$, then $z_1 = z_2$.

(25) If $z_1 + z = z_2 + z$ or $z_1 + z = z + z_2$, then $z_1 = z_2$.

(26) $-(z_1 + z_2) = -z_1 + -z_2$.

(27) $(-z_1) \cdot z_2 = -z_1 \cdot z_2$ and $z_1 \cdot -z_2 = -z_1 \cdot z_2$.

(28) $(-z_1) \cdot -z_2 = z_1 \cdot z_2$.

(29) $-z = (-\text{the unity of } \mathbb{C}_{\mathrm{F}}) \cdot z$.

(30) $z_1 - z_2 = z_1 + -z_2$.

(31) If $z_1 - z_2 =$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $z_1 = z_2$.

(32) $z - z =$ the zero of $\mathbb{C}_{\mathrm{F}}$.

(33) $z -$ the zero of $\mathbb{C}_{\mathrm{F}} = z$.

(34) (The zero of $\mathbb{C}_{\mathrm{F}}) - z = -z$.

(35) $z_1 - -z_2 = z_1 + z_2$.

(36) $-(z_1 - z_2) = -z_1 + z_2$.

(37) $-(z_1 - z_2) = z_2 - z_1$.

(38) $z_1 + (z_2 - z_3) = (z_1 + z_2) - z_3$.

(39) $z_1 - (z_2 - z_3) = (z_1 - z_2) + z_3$.

(40) $z_1 - z_2 - z_3 = z_1 - (z_2 + z_3)$.

(41) $z_1 = (z_1 + z) - z$.

(42) $z_1 = (z_1 - z) + z$.

(43) $z \cdot (z_1 - z_2) = z \cdot z_1 - z \cdot z_2$ and $(z_1 - z_2) \cdot z = z_1 \cdot z - z_2 \cdot z$.

(44) If $z \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $z \cdot z^{-1} =$ the unity of $\mathbb{C}_{\mathrm{F}}$ and $z^{-1} \cdot z =$ the unity of $\mathbb{C}_{\mathrm{F}}$.

(45) If $z_1 \cdot z_2 =$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $z_1 =$ the zero of $\mathbb{C}_{\mathrm{F}}$ or $z_2 =$ the zero of $\mathbb{C}_{\mathrm{F}}$.

(46) If $z \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $z^{-1} \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$.

(47) If $z_1 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and ${z_1}^{-1} = {z_2}^{-1}$, then $z_1 = z_2$.

(48) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and if $z_1 \cdot z_2 =$ the unity of $\mathbb{C}_{\mathrm{F}}$ or $z_2 \cdot z_1 =$ the unity of $\mathbb{C}_{\mathrm{F}}$, then $z_1 = {z_2}^{-1}$.

(49) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and if $z_1 \cdot z_2 = z_3$ or $z_2 \cdot z_1 = z_3$, then $z_1 = z_3 \cdot {z_2}^{-1}$ and $z_1 = {z_2}^{-1} \cdot z_3$.

(50) (The unity of $\mathbb{C}_{\mathrm{F}})^{-1} =$ the unity of $\mathbb{C}_{\mathrm{F}}$.

(51) If $z_1 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $(z_1 \cdot z_2)^{-1} = {z_1}^{-1} \cdot {z_2}^{-1}$.

(52) If $z \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $(z^{-1})^{-1} = z$.

(53) If $z \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $(-z)^{-1} = -z^{-1}$.

(54) If $z \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and if $z_1 \cdot z = z_2 \cdot z$ or $z_1 \cdot z = z \cdot z_2$, then $z_1 = z_2$.

(55) If $z_1 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then ${z_1}^{-1} + {z_2}^{-1} = (z_1 + z_2) \cdot (z_1 \cdot z_2)^{-1}$.

(56) If $z_1 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then ${z_1}^{-1} - {z_2}^{-1} = (z_2 - z_1) \cdot (z_1 \cdot z_2)^{-1}$.

(57) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{z_1}{z_2} = z_1 \cdot {z_2}^{-1}$.

(58) If $z \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $z^{-1} = \frac{\text{the unity of } \mathbb{C}_{\mathrm{F}}}{z}$.

(59) $\frac{z}{\text{the unity of } \mathbb{C}_{\mathrm{F}}} = z$.

(60) If $z \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{z}{z} =$ the unity of $\mathbb{C}_{\mathrm{F}}$.

(61) If $z \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{\text{the zero of } \mathbb{C}_{\mathrm{F}}}{z} =$ the zero of $\mathbb{C}_{\mathrm{F}}$.

(62) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $\frac{z_1}{z_2} =$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $z_1 =$ the zero of $\mathbb{C}_{\mathrm{F}}$.

(63) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_4 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{z_1}{z_2} \cdot \frac{z_3}{z_4} = \frac{z_1 \cdot z_3}{z_2 \cdot z_4}$.

(64) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $z \cdot \frac{z_1}{z_2} = \frac{z \cdot z_1}{z_2}$.

(65) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $\frac{z_1}{z_2} =$ the unity of $\mathbb{C}_{\mathrm{F}}$, then $z_1 = z_2$.

(66) If $z \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $z_1 = \frac{z_1 \cdot z}{z}$.

(67) If $z_1 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $(\frac{z_1}{z_2})^{-1} = \frac{z_2}{z_1}$.

(68) If $z_1 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{{z_1}^{-1}}{{z_2}^{-1}} = \frac{z_2}{z_1}$.

(69) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{z_1}{{z_2}^{-1}} = z_1 \cdot z_2$.

(70) If $z_1 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{{z_1}^{-1}}{z_2} = (z_1 \cdot z_2)^{-1}$.

(71) If $z_1 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then ${z_1}^{-1} \cdot \frac{z}{z_2} = \frac{z}{z_1 \cdot z_2}$.

(72) If $z \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{z_1}{z_2} = \frac{z_1 \cdot z}{z_2 \cdot z}$ and $\frac{z_1}{z_2} = \frac{z \cdot z_1}{z \cdot z_2}$.

(73) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_3 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{z_1}{z_2 \cdot z_3} = \frac{\frac{z_1}{z_2}}{z_3}$.

(74) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_3 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{z_1 \cdot z_3}{z_2} = \frac{z_1}{\frac{z_2}{z_3}}$.

(75) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_3 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_4 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{\frac{z_1}{z_2}}{\frac{z_3}{z_4}} = \frac{z_1 \cdot z_4}{z_2 \cdot z_3}$.

(76) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_4 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{z_1}{z_2} + \frac{z_3}{z_4} = \frac{z_1 \cdot z_4 + z_3 \cdot z_2}{z_2 \cdot z_4}$.

(77) If $z \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{z_1}{z} + \frac{z_2}{z} = \frac{z_1 + z_2}{z}$.

(78) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $-\frac{z_1}{z_2} = \frac{-z_1}{z_2}$ and $-\frac{z_1}{z_2} = \frac{z_1}{-z_2}$.

(79) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{z_1}{z_2} = \frac{-z_1}{-z_2}$.

(80) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and $z_4 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{z_1}{z_2} - \frac{z_3}{z_4} = \frac{z_1 \cdot z_4 - z_3 \cdot z_2}{z_2 \cdot z_4}$.

(81) If $z \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $\frac{z_1}{z} - \frac{z_2}{z} = \frac{z_1 - z_2}{z}$.

(82) If $z_2 \neq$ the zero of $\mathbb{C}_{\mathrm{F}}$ and if $z_1 \cdot z_2 = z_3$ or $z_2 \cdot z_1 = z_3$, then $z_1 = \frac{z_3}{z_2}$.

(83) (the zero of $\mathbb{C}_{\mathrm{F}})^* =$ the zero of $\mathbb{C}_{\mathrm{F}}$.

(84) If $z^* =$ the zero of $\mathbb{C}_{\mathrm{F}}$, then $z =$ the zero of $\mathbb{C}_{\mathrm{F}}$.

(85) (the unity of $\mathbb{C}_{\mathrm{F}})^* =$ the unity of $\mathbb{C}_{\mathrm{F}}$.

(86) $(z^*)^* = z$.

(87) $(z_1 + z_2)^* = {z_1}^* + {z_2}^*$.

(88) $(-z)^* = -z^*$.

(89) $(z_1 - z_2)^* = {z_1}^* - {z_2}^*$.

(90) $(z_1 \cdot z_2)^* = {z_1}^* \cdot {z_2}^*$.

(91) If $z \neq$ the zero of $\mathbb{C}_{\rm F}$, then $(z^{-1})^* = (z^*)^{-1}$.

(92) If $z_2 \neq$ the zero of $\mathbb{C}_{\rm F}$, then $(\frac{z_1}{z_2})^* = \frac{{z_1}^*}{{z_2}^*}$.

(93) $|$the zero of $\mathbb{C}_{\rm F}| = 0$.

(94) If $|z| = 0$, then $z =$ the zero of $\mathbb{C}_{\rm F}$.

(95) $0 \leqslant |z|$.

(96) $z \neq$ the zero of $\mathbb{C}_{\rm F}$ iff $0 < |z|$.

(97) $|$the unity of $\mathbb{C}_{\rm F}| = 1$.

(98) $|-z| = |z|$.

(99) $|z^*| = |z|$.

(100) $|z_1 + z_2| \leqslant |z_1| + |z_2|$.

(101) $|z_1 - z_2| \leqslant |z_1| + |z_2|$.

(102) $|z_1| - |z_2| \leqslant |z_1 + z_2|$.

(103) $|z_1| - |z_2| \leqslant |z_1 - z_2|$.

(104) $|z_1 - z_2| = |z_2 - z_1|$.

(105) $|z_1 - z_2| = 0$ iff $z_1 = z_2$.

(106) $z_1 \neq z_2$ iff $0 < |z_1 - z_2|$.

(107) $|z_1 - z_2| \leqslant |z_1 - z| + |z - z_2|$.

(108) $||z_1| - |z_2|| \leqslant |z_1 - z_2|$.

(109) $|z_1 \cdot z_2| = |z_1| \cdot |z_2|$.

(110) If $z \neq$ the zero of $\mathbb{C}_{\rm F}$, then $|z^{-1}| = |z|^{-1}$.

(111) If $z_2 \neq$ the zero of $\mathbb{C}_{\rm F}$, then $\frac{|z_1|}{|z_2|} = |\frac{z_1}{z_2}|$.

(112) $|z \cdot z| = |z \cdot z^*|$.

References

[1] Czesław Byliński. The complex numbers. *Formalized Mathematics*, 1(**3**):507–513, 1990.

[2] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[3] Eugeniusz Kusak, Wojciech Leończuk, and Michał Muzalewski. Abelian groups, fields and vector spaces. *Formalized Mathematics*, 1(**2**):335–342, 1990.

[4] Jan Popiołek. Some properties of functions modul and signum. *Formalized Mathematics*, 1(**2**):263–264, 1990.

[5] Wojciech A. Trybulec. Vectors in real linear space. *Formalized Mathematics*, 1(**2**):291–296, 1990.

[6] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

Received January 18, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Integrability of Bounded Total Functions

Noboru Endou
Shinshu University
Nagano

Katsumi Wasaki
Shinshu University
Nagano

Yasunari Shidama
Shinshu University
Nagano

Summary. All these results have been obtained by Darboux's theorem in our previous article [7]. In addition, we have proved the first mean value theorem to Riemann integral.

MML Identifier: INTEGRA4.

The articles [15], [1], [2], [3], [6], [8], [4], [5], [9], [18], [12], [14], [13], [11], [10], [17], and [16] provide the notation and terminology for this paper.

1. Basic Integrable Functions and First Mean Value Theorem

For simplicity, we use the following convention: i, n denote natural numbers, a, r, x, y denote real numbers, A denotes a closed-interval subset of $\mathbb{R}$, C denotes a non empty set, and X denotes a set.

We now state several propositions:

(1) For every element D of divs A such that $\mathrm{vol}(A) = 0$ holds $\mathrm{len}\, D = 1$.

(2) $\chi_{A,A}$ is integrable on A and $\mathrm{integral}\, \chi_{A,A} = \mathrm{vol}(A)$.

(3) For every partial function f from A to $\mathbb{R}$ and for every r holds f is total and $\mathrm{rng}\, f = \{r\}$ iff $f = r\, \chi_{A,A}$.

(4) Let f be a partial function from A to $\mathbb{R}$ and given r. If f is total and $\mathrm{rng}\, f = \{r\}$, then f is integrable on A and $\mathrm{integral}\, f = r \cdot \mathrm{vol}(A)$.

(5) For every r there exists a partial function f from A to $\mathbb{R}$ such that f is total and $\mathrm{rng}\, f = \{r\}$ and f is bounded on A.

(6) Let f be a partial function from A to $\mathbb{R}$ and D be an element of divs A. If $\mathrm{vol}(A) = 0$, then f is integrable on A and $\mathrm{integral}\, f = 0$.

(7) Let f be a partial function from A to $\mathbb{R}$. Suppose f is total and bounded on A and f is integrable on A. Then there exists a such that $\inf \operatorname{rng} f \leqslant a$ and $a \leqslant \sup \operatorname{rng} f$ and $\operatorname{integral} f = a \cdot \operatorname{vol}(A)$.

2. Integrability of Bounded Total Functions

We now state three propositions:

(8) Let f be a partial function from A to $\mathbb{R}$ and T be a DivSequence of A. Suppose f is total and bounded on A and δ_T is convergent and $\lim(\delta_T) = 0$. Then $\text{lower\_sum}(f, T)$ is convergent and $\lim \text{lower\_sum}(f, T) = \text{lower\_integral}\, f$.

(9) Let f be a partial function from A to $\mathbb{R}$ and T be a DivSequence of A. Suppose f is total and bounded on A and δ_T is convergent and $\lim(\delta_T) = 0$. Then $\text{upper\_sum}(f, T)$ is convergent and $\lim \text{upper\_sum}(f, T) = \text{upper\_integral}\, f$.

(10) Let f be a partial function from A to $\mathbb{R}$. Suppose f is total and bounded on A. Then f is upper integrable on A and f is lower integrable on A.

Let A be a closed-interval subset of $\mathbb{R}$, let I_1 be an element of $\operatorname{divs} A$, and let us consider n. We say that I_1 divides into equal n if and only if:

(Def. 1) $\operatorname{len} I_1 = n$ and for every i such that $i \in \operatorname{dom} I_1$ holds $I_1(i) = \inf A + \frac{\operatorname{vol}(A)}{\operatorname{len} I_1} \cdot i$.

Next we state a number of propositions:

(11) There exists a DivSequence T of A such that δ_T is convergent and $\lim(\delta_T) = 0$.

(12) Let f be a partial function from A to $\mathbb{R}$. Suppose f is total and bounded on A. Then f is integrable on A if and only if for every DivSequence T of A such that δ_T is convergent and $\lim(\delta_T) = 0$ holds $\lim \text{upper\_sum}(f, T) - \lim \text{lower\_sum}(f, T) = 0$.

(13) For every partial function f from C to $\mathbb{R}$ such that f is total holds $\max_+(f)$ is total and $\max_-(f)$ is total.

(14) For every partial function f from C to $\mathbb{R}$ such that f is upper bounded on X holds $\max_+(f)$ is upper bounded on X.

(15) For every partial function f from C to $\mathbb{R}$ holds $\max_+(f)$ is lower bounded on X.

(16) For every partial function f from C to $\mathbb{R}$ such that f is lower bounded on X holds $\max_-(f)$ is upper bounded on X.

(17) For every partial function f from C to $\mathbb{R}$ holds $\max_-(f)$ is lower bounded on X.

(18) For every partial function f from A to $\mathbb{R}$ such that f is upper bounded on A holds $\operatorname{rng}(f\restriction X)$ is upper bounded.

(19) For every partial function f from A to $\mathbb{R}$ such that f is lower bounded on A holds $\operatorname{rng}(f\restriction X)$ is lower bounded.

(20) Let f be a partial function from A to $\mathbb{R}$. Suppose f is total and bounded on A and f is integrable on A. Then $\max_{+}(f)$ is integrable on A.

(21) For every partial function f from C to $\mathbb{R}$ holds $\max_{-}(f) = \max_{+}(-f)$.

(22) Let f be a partial function from A to $\mathbb{R}$. Suppose f is total and bounded on A and f is integrable on A. Then $\max_{-}(f)$ is integrable on A.

(23) Let f be a partial function from A to $\mathbb{R}$. Suppose f is total and bounded on A and f is integrable on A. Then $|f|$ is integrable on A and $|\operatorname{integral} f| \leqslant \operatorname{integral} |f|$.

(24) Let f be a partial function from A to $\mathbb{R}$. Suppose f is bounded on A and total and for all x, y such that $x \in A$ and $y \in A$ holds $|f(x) - f(y)| \leqslant a$. Then $\sup \operatorname{rng} f - \inf \operatorname{rng} f \leqslant a$.

(25) Let f, g be partial functions from A to $\mathbb{R}$. Suppose that
(i) f is bounded on A,
(ii) g is bounded on A,
(iii) f is total,
(iv) g is total,
(v) $a \geqslant 0$, and
(vi) for all x, y such that $x \in A$ and $y \in A$ holds $|g(x) - g(y)| \leqslant a \cdot |f(x) - f(y)|$.
Then $\sup \operatorname{rng} g - \inf \operatorname{rng} g \leqslant a \cdot (\sup \operatorname{rng} f - \inf \operatorname{rng} f)$.

(26) Let f, g, h be partial functions from A to $\mathbb{R}$. Suppose that f is bounded on A and g is bounded on A and h is bounded on A and f is total and g is total and h is total and $a \geqslant 0$ and for all x, y such that $x \in A$ and $y \in A$ holds $|h(x) - h(y)| \leqslant a \cdot (|f(x) - f(y)| + |g(x) - g(y)|)$. Then $\sup \operatorname{rng} h - \inf \operatorname{rng} h \leqslant a \cdot ((\sup \operatorname{rng} f - \inf \operatorname{rng} f) + (\sup \operatorname{rng} g - \inf \operatorname{rng} g))$.

(27) Let f, g be partial functions from A to $\mathbb{R}$. Suppose that
(i) f is total and bounded on A,
(ii) f is integrable on A,
(iii) g is total and bounded on A,
(iv) $a > 0$, and
(v) for all x, y such that $x \in A$ and $y \in A$ holds $|g(x) - g(y)| \leqslant a \cdot |f(x) - f(y)|$.
Then g is integrable on A.

(28) Let f, g, h be partial functions from A to $\mathbb{R}$. Suppose that f is total and bounded on A and f is integrable on A and g is total and bounded on A and g is integrable on A and h is total and bounded on A and

$a > 0$ and for all x, y such that $x \in A$ and $y \in A$ holds $|h(x) - h(y)| \leqslant a \cdot (|f(x) - f(y)| + |g(x) - g(y)|)$. Then h is integrable on A.

(29) Let f, g be partial functions from A to $\mathbb{R}$. Suppose that

(i) f is total and bounded on A,

(ii) f is integrable on A,

(iii) g is total and bounded on A, and

(iv) g is integrable on A.

Then $f\,g$ is integrable on A.

(30) Let f be a partial function from A to $\mathbb{R}$. Suppose f is total and bounded on A and f is integrable on A and $0 \notin \operatorname{rng} f$ and $\frac{1}{f}$ is bounded on A. Then $\frac{1}{f}$ is integrable on A.

References

[1] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[2] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[3] Czesław Byliński. Functions from a set to a set. *Formalized Mathematics*, 1(**1**):153–164, 1990.

[4] Czesław Byliński. Partial functions. *Formalized Mathematics*, 1(**2**):357–367, 1990.

[5] Czesław Byliński and Piotr Rudnicki. Bounding boxes for compact sets in $\mathcal{E}^2$. *Formalized Mathematics*, 6(**3**):427–440, 1997.

[6] Noboru Endou and Artur Korniłowicz. The definition of the Riemann definite integral and some related lemmas. *Formalized Mathematics*, 8(**1**):93–102, 1999.

[7] Noboru Endou, Katsumi Wasaki, and Yasunari Shidama. Darboux's theorem. *Formalized Mathematics*, 9(**1**):197–200, 2001.

[8] Noboru Endou, Katsumi Wasaki, and Yasunari Shidama. Scalar multiple of Riemann definite integral. *Formalized Mathematics*, 9(**1**):191–196, 2001.

[9] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[10] Jarosław Kotowicz. Convergent real sequences. Upper and lower bound of sets of real numbers. *Formalized Mathematics*, 1(**3**):477–481, 1990.

[11] Jarosław Kotowicz. Convergent sequences and the limit of sequences. *Formalized Mathematics*, 1(**2**):273–275, 1990.

[12] Jarosław Kotowicz. Partial functions from a domain to the set of real numbers. *Formalized Mathematics*, 1(**4**):703–709, 1990.

[13] Jarosław Kotowicz. Real sequences and basic operations on them. *Formalized Mathematics*, 1(**2**):269–272, 1990.

[14] Jarosław Kotowicz and Yuji Sakai. Properties of partial functions from a domain to the set of real numbers. *Formalized Mathematics*, 3(**2**):279–288, 1992.

[15] Jan Popiołek. Some properties of functions modul and signum. *Formalized Mathematics*, 1(**2**):263–264, 1990.

[16] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[17] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[18] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

Received February 1, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

High-Speed Algorithms for RSA Cryptograms

Yasushi Fuwa
Shinshu University
Nagano

Yoshinori Fujisawa
Shinshu University
Nagano

Summary. In this article, we propose a new high-speed processing method for encoding and decoding the RSA cryptogram that is a kind of public-key cryptogram. This cryptogram is not only used for encrypting data, but also for such purposes as authentication. However, the encoding and decoding processes take a long time because they require a great deal of calculations. As a result, this cryptogram is not suited for practical use. Until now, we proposed a high-speed algorithm of addition using radix-2^k signed-digit numbers and clarified correctness of it ([5]). In this article, we defined two new operations for a high-speed coding and encoding processes on public-key cryptograms based on radix-2^k signed-digit (SD) numbers. One is calculation of $(a*b) \bmod c$ (a, b, c are natural numbers). Another one is calculation of $(a^b) \bmod c$ (a, b, c are natural numbers). Their calculations are realized repetition of addition. We propose a high-speed algorithm of their calculations using proposed addition algorithm and clarify the correctness of them. In the first section, we prepared some useful theorems for natural numbers and integers and so on. In the second section, we proved some properties of addition operation using a radix-2^k SD numbers. In the third section, we defined some functions on the relation between a finite sequence of k-SD and a finite sequence of $\mathbb{N}$ and proved some properties about them. In the fourth section, algorithm of calculation of $(a * b) \bmod c$ based on radix-2^k SD numbers is proposed and its correctness is clarified. In the last section, algorithm of calculation of $(a^b) \bmod c$ based on radix-2^k SD numbers is proposed and we clarified its correctness.

MML Identifier: RADIX_2.

The notation and terminology used in this paper are introduced in the following articles: [8], [6], [2], [3], [4], [9], [1], [5], [10], and [7].

1. Some Useful Theorems

In this paper k is a natural number.

The following propositions are true:

(1) For every natural number a holds $a \bmod 1 = 0$.

(2) Let a, b be integers and n be a natural number. If $n > 0$, then $((a \bmod n) + (b \bmod n)) \bmod n = (a + (b \bmod n)) \bmod n$ and $((a \bmod n) + (b \bmod n)) \bmod n = ((a \bmod n) + b) \bmod n$.

(3) For all integers a, b and for every natural number n such that $n > 0$ holds $a \cdot b \bmod n = a \cdot (b \bmod n) \bmod n$ and $a \cdot b \bmod n = (a \bmod n) \cdot b \bmod n$.

(4) For all natural numbers a, b, i such that $1 \leqslant i$ and $0 < b$ holds $(a \bmod b^{i}_{\mathbb{N}}) \div b^{i -' 1}_{\mathbb{N}} = (a \div b^{i -' 1}_{\mathbb{N}}) \bmod b$.

(5) For all natural numbers i, n such that $i \in \mathrm{Seg}\, n$ holds $i + 1 \in \mathrm{Seg}(n+1)$.

2. Properties of Addition Operation Using Radix-2^k Signed-Digit Numbers

One can prove the following propositions:

(6) For every natural number k holds $\mathrm{Radix}\, k > 0$.

(7) For every tuple x of 1 and k $-$SD holds $\mathrm{SDDec}\, x = \mathrm{DigA}(x, 1)$.

(8) For every integer x holds $\mathrm{SD\_Add\_Data}(x, k) + \mathrm{SD\_Add\_Carry}\, x \cdot \mathrm{Radix}\, k = x$.

(9) Let n be a natural number, x be a tuple of $n+1$ and k $-$SD, and x_1 be a tuple of n and k $-$SD. Suppose that for every natural number i such that $i \in \mathrm{Seg}\, n$ holds $x(i) = x_1(i)$. Then $\sum \mathrm{DigitSD}\, x = \sum((\mathrm{DigitSD}\, x_1) \frown \langle \mathrm{SubDigit}(x, n+1, k) \rangle)$.

(10) Let n be a natural number, x be a tuple of $n+1$ and k $-$SD, and x_1 be a tuple of n and k $-$SD. Suppose that for every natural number i such that $i \in \mathrm{Seg}\, n$ holds $x(i) = x_1(i)$. Then $\mathrm{SDDec}\, x_1 + ((\mathrm{Radix}\, k)^{n}_{\mathbb{N}}) \cdot \mathrm{DigA}(x, n+1) = \mathrm{SDDec}\, x$.

(11) Let n be a natural number. Suppose $n \geqslant 1$. Let x, y be tuples of n and k $-$SD. If $k \geqslant 2$, then $\mathrm{SDDec}\, x' +' y + \mathrm{SD\_Add\_Carry}\, \mathrm{DigA}(x, n) + \mathrm{DigA}(y, n) \cdot (\mathrm{Radix}\, k)^{n}_{\mathbb{N}} = \mathrm{SDDec}\, x + \mathrm{SDDec}\, y$.

3. Definitions on the Relation Between a Finite Sequence of k-SD and a Finite Sequence of $\mathbb{N}$ and Some Properties about them

Let i, k, n be natural numbers and let x be a tuple of n and $\mathbb{N}$. The functor $\mathrm{SubDigit2}(x,i,k)$ yielding an element of $\mathbb{N}$ is defined by:

(Def. 1) $\mathrm{SubDigit2}(x,i,k) = ((\mathrm{Radix}\,k)_{\mathbb{N}}^{i-'1}) \cdot x(i)$.

Let n, k be natural numbers and let x be a tuple of n and $\mathbb{N}$. The functor $\mathrm{DigitSD2}(x,k)$ yields a tuple of n and $\mathbb{N}$ and is defined as follows:

(Def. 2) For every natural number i such that $i \in \mathrm{Seg}\,n$ holds $\pi_i\,\mathrm{DigitSD2}(x,k) = \mathrm{SubDigit2}(x,i,k)$.

Let n, k be natural numbers and let x be a tuple of n and $\mathbb{N}$. The functor $\mathrm{SDDec2}(x,k)$ yielding a natural number is defined as follows:

(Def. 3) $\mathrm{SDDec2}(x,k) = \sum \mathrm{DigitSD2}(x,k)$.

Let i, k, x be natural numbers. The functor $\mathrm{DigitDC2}(x,i,k)$ yields a natural number and is defined as follows:

(Def. 4) $\mathrm{DigitDC2}(x,i,k) = (x \bmod (\mathrm{Radix}\,k)_{\mathbb{N}}^{i}) \div (\mathrm{Radix}\,k)_{\mathbb{N}}^{i-'1}$.

Let k, n, x be natural numbers. The functor $\mathrm{DecSD2}(x,n,k)$ yielding a tuple of n and $\mathbb{N}$ is defined by:

(Def. 5) For every natural number i such that $i \in \mathrm{Seg}\,n$ holds $(\mathrm{DecSD2}(x,n,k))(i) = \mathrm{DigitDC2}(x,i,k)$.

The following propositions are true:

(12) Let n, k be natural numbers, x be a tuple of n and $\mathbb{N}$, and y be a tuple of n and k −SD. If $x = y$, then $\mathrm{DigitSD2}(x,k) = \mathrm{DigitSD}\,y$.

(13) Let n, k be natural numbers, x be a tuple of n and $\mathbb{N}$, and y be a tuple of n and k −SD. If $x = y$, then $\mathrm{SDDec2}(x,k) = \mathrm{SDDec}\,y$.

(14) For all natural numbers x, n, k holds $\mathrm{DecSD2}(x,n,k) = \mathrm{DecSD}(x,n,k)$.

(15) Let n be a natural number. Suppose $n \geqslant 1$. Let m, k be natural numbers. If m is represented by n, k, then $m = \mathrm{SDDec2}(\mathrm{DecSD2}(m,n,k),k)$.

4. A High-Speed Algorithm of Calculation of $(a * b)$ mod b Based on Radix-2^k Signed-Digit Numbers and its Correctness

Let q be an integer, let f, j, k, n be natural numbers, and let c be a tuple of n and k −SD. The functor $\mathrm{Table1}(q,c,f,j)$ yielding an integer is defined as follows:

(Def. 6) $\mathrm{Table1}(q,c,f,j) = q \cdot \mathrm{DigA}(c,j) \bmod f$.

Let q be an integer, let k, f, n be natural numbers, and let c be a tuple of n and k −SD. Let us assume that $n \geqslant 1$. The functor Mul_mod(q, c, f, k) yielding a tuple of n and $\mathbb{Z}$ is defined by the conditions (Def. 7).

(Def. 7)(i) (Mul_mod$(q, c, f, k))(1) = \mathrm{Table1}(q, c, f, n)$, and

(ii) for every natural number i such that $1 \leqslant i$ and $i \leqslant n - 1$ there exist integers I_1, I_2 such that $I_1 = (\mathrm{Mul\_mod}(q, c, f, k))(i)$ and $I_2 = (\mathrm{Mul\_mod}(q, c, f, k))(i + 1)$ and $I_2 = (\mathrm{Radix}\,k \cdot I_1 + \mathrm{Table1}(q, c, f, n -' i)) \bmod f$.

One can prove the following proposition

(16) Let n be a natural number. Suppose $n \geqslant 1$. Let q be an integer and i_1, f, k be natural numbers. Suppose i_1 is represented by n, k and $f > 0$. Let c be a tuple of n and k −SD. If $c = \mathrm{DecSD}(i_1, n, k)$, then $(\mathrm{Mul\_mod}(q, c, f, k))(n) = q \cdot i_1 \bmod f$.

5. A High-Speed Algorithm of Calculation of (a^b) mod b Based on a Radix-2^k Signed-Digit Numbers and its Correctness

Let n, f, j, m be natural numbers and let e be a tuple of n and $\mathbb{N}$. The functor Table2(m, e, f, j) yielding a natural number is defined as follows:

(Def. 8) $\mathrm{Table2}(m, e, f, j) = (m_{\mathbb{N}}^{\pi_j e}) \bmod f$.

Let k, f, m, n be natural numbers and let e be a tuple of n and $\mathbb{N}$. Let us assume that $n \geqslant 1$. The functor Pow_mod(m, e, f, k) yields a tuple of n and $\mathbb{N}$ and is defined by the conditions (Def. 9).

(Def. 9)(i) $(\mathrm{Pow\_mod}(m, e, f, k))(1) = \mathrm{Table2}(m, e, f, n)$, and

(ii) for every natural number i such that $1 \leqslant i$ and $i \leqslant n - 1$ there exist natural numbers i_2, i_3 such that $i_2 = (\mathrm{Pow\_mod}(m, e, f, k))(i)$ and $i_3 = (\mathrm{Pow\_mod}(m, e, f, k))(i+1)$ and $i_3 = ((i2_{\mathbb{N}}^{\mathrm{Radix}\,k})\mathrm{mod} f) \cdot \mathrm{Table2}(m, e, f, n -' i) \bmod f$.

One can prove the following proposition

(17) Let n be a natural number. Suppose $n \geqslant 1$. Let m, k, f, i_4 be natural numbers. Suppose i_4 is represented by n, k and $f > 0$. Let e be a tuple of n and $\mathbb{N}$. If $e = \mathrm{DecSD2}(i_4, n, k)$, then $(\mathrm{Pow_mod}(m, e, f, k))(n) = (m_{\mathbb{N}}^{i_4})$mod f.

References

[1] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[2] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[3] Czesław Byliński. Finite sequences and tuples of elements of a non-empty sets. *Formalized Mathematics*, 1(**3**):529–536, 1990.

[4] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[5] Yoshinori Fujisawa and Yasushi Fuwa. Definitions of radix-2^k signed-digit number and its adder algorithm. *Formalized Mathematics*, 9(**1**):71–75, 2001.

[6] Yoshinori Fujisawa, Yasushi Fuwa, and Hidetaka Shimizu. Euler's Theorem and small Fermat's Theorem. *Formalized Mathematics*, 7(**1**):123–126, 1998.

[7] Andrzej Kondracki. The Chinese Remainder Theorem. *Formalized Mathematics*, 6(**4**):573–577, 1997.

[8] Takaya Nishiyama and Yasuho Mizuhara. Binary arithmetics. *Formalized Mathematics*, 4(**1**):83–86, 1993.

[9] Michał J. Trybulec. Integers. *Formalized Mathematics*, 1(**3**):501–505, 1990.

[10] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

Received February 1, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Definition of Integrability for Partial Functions from $\mathbb{R}$ to $\mathbb{R}$ and Integrability for Continuous Functions

Noboru Endou
Shinshu University
Nagano

Katsumi Wasaki
Shinshu University
Nagano

Yasunari Shidama
Shinshu University
Nagano

Summary. In this article, we defined the Riemann definite integral of partial function from $\mathbb{R}$ to $\mathbb{R}$. Then we have proved the integrability for the continuous function and differentiable function. Moreover, we have proved an elementary theorem of calculus.

MML Identifier: `INTEGRA5`.

The articles [12], [13], [1], [2], [6], [3], [5], [14], [7], [16], [9], [10], [4], [11], [8], and [15] provide the notation and terminology for this paper.

1. Some Useful Properties of Finite Sequence

For simplicity, we adopt the following convention: i denotes a natural number, a, b, r_1, r_2 denote real numbers, A denotes a closed-interval subset of $\mathbb{R}$, C denotes a non empty set, and X denotes a set.

One can prove the following propositions:

(1) Let F, F_1, F_2 be finite sequences of elements of $\mathbb{R}$ and given r_1, r_2. If $F_1 = \langle r_1 \rangle \frown F$ or $F_1 = F \frown \langle r_1 \rangle$ and if $F_2 = \langle r_2 \rangle \frown F$ or $F_2 = F \frown \langle r_2 \rangle$, then $\sum (F_1 - F_2) = r_1 - r_2$.

(2) Let F_1, F_2 be finite sequences of elements of $\mathbb{R}$. If $\operatorname{len} F_1 = \operatorname{len} F_2$, then $\operatorname{len}(F_1 + F_2) = \operatorname{len} F_1$ and $\operatorname{len}(F_1 - F_2) = \operatorname{len} F_1$ and $\sum (F_1 + F_2) = \sum F_1 + \sum F_2$ and $\sum (F_1 - F_2) = \sum F_1 - \sum F_2$.

(3) Let F_1, F_2 be finite sequences of elements of $\mathbb{R}$. If $\operatorname{len} F_1 = \operatorname{len} F_2$ and for every i such that $i \in \operatorname{dom} F_1$ holds $F_1(i) \leqslant F_2(i)$, then $\sum F_1 \leqslant \sum F_2$.

2. Integrability for Partial Function of $\mathbb{R}$, $\mathbb{R}$

Let C be a non empty subset of $\mathbb{R}$ and let f be a partial function from $\mathbb{R}$ to $\mathbb{R}$. The functor $f \restriction C$ yielding a partial function from C to $\mathbb{R}$ is defined as follows:

(Def. 1) $f \restriction C = f{\restriction}C$.

Next we state two propositions:

(4) For all partial functions f, g from $\mathbb{R}$ to $\mathbb{R}$ and for every non empty subset C of $\mathbb{R}$ holds $(f \restriction C)\,(g \restriction C) = (f\,g) \restriction C$.

(5) For all partial functions f, g from $\mathbb{R}$ to $\mathbb{R}$ and for every non empty subset C of $\mathbb{R}$ holds $(f + g) \restriction C = f \restriction C + g \restriction C$.

Let A be a closed-interval subset of $\mathbb{R}$ and let f be a partial function from $\mathbb{R}$ to $\mathbb{R}$. We say that f is integrable on A if and only if:

(Def. 2) $f \restriction A$ is integrable on A.

Let A be a closed-interval subset of $\mathbb{R}$ and let f be a partial function from $\mathbb{R}$ to $\mathbb{R}$. The functor $\int\limits_A f(x)dx$ yields a real number and is defined by:

(Def. 3) $\int\limits_A f(x)dx = \mathrm{integral}\, f \restriction A$.

The following propositions are true:

(6) For every partial function f from $\mathbb{R}$ to $\mathbb{R}$ such that $A \subseteq \mathrm{dom}\, f$ holds $f \restriction A$ is total.

(7) For every partial function f from $\mathbb{R}$ to $\mathbb{R}$ such that f is upper bounded on A holds $f \restriction A$ is upper bounded on A.

(8) For every partial function f from $\mathbb{R}$ to $\mathbb{R}$ such that f is lower bounded on A holds $f \restriction A$ is lower bounded on A.

(9) For every partial function f from $\mathbb{R}$ to $\mathbb{R}$ such that f is bounded on A holds $f \restriction A$ is bounded on A.

3. Integrability for Continuous Function

The following propositions are true:

(10) For every partial function f from $\mathbb{R}$ to $\mathbb{R}$ such that f is continuous on A holds f is bounded on A.

(11) For every partial function f from $\mathbb{R}$ to $\mathbb{R}$ such that f is continuous on A holds f is integrable on A.

(12) Let f be a partial function from $\mathbb{R}$ to $\mathbb{R}$ and D be an element of $\mathrm{divs}\, A$. Suppose $A \subseteq X$ and f is differentiable on X and $f'_{\restriction X}$ is bounded on A. Then $\mathrm{lower\_sum}(f'_{\restriction X} \restriction A, D) \leqslant f(\sup A) - f(\inf A)$ and $f(\sup A) - f(\inf A) \leqslant \mathrm{upper\_sum}(f'_{\restriction X} \restriction A, D)$.

(13) Let f be a partial function from $\mathbb{R}$ to $\mathbb{R}$. Suppose $A \subseteq X$ and f is differentiable on X and $f'_{\restriction X}$ is integrable on A and $f'_{\restriction X}$ is bounded on A. Then $\int\limits_A f'_{\restriction X}(x)dx = f(\sup A) - f(\inf A)$.

(14) For every partial function f from $\mathbb{R}$ to $\mathbb{R}$ such that f is non-decreasing on A and $A \subseteq \operatorname{dom} f$ holds $\operatorname{rng}(f \restriction A)$ is bounded.

(15) Let f be a partial function from $\mathbb{R}$ to $\mathbb{R}$. If f is non-decreasing on A and $A \subseteq \operatorname{dom} f$, then $\inf \operatorname{rng}(f \restriction A) = f(\inf A)$ and $\sup \operatorname{rng}(f \restriction A) = f(\sup A)$.

(16) For every partial function f from $\mathbb{R}$ to $\mathbb{R}$ such that f is monotone on A and $A \subseteq \operatorname{dom} f$ holds f is integrable on A.

(17) Let f be a partial function from $\mathbb{R}$ to $\mathbb{R}$ and A, B be closed-interval subsets of $\mathbb{R}$. If f is continuous on A and $B \subseteq A$, then f is integrable on B.

(18) Let f be a partial function from $\mathbb{R}$ to $\mathbb{R}$, A, B, C be closed-interval subsets of $\mathbb{R}$, and given X. Suppose $A \subseteq X$ and f is differentiable on X and $f'_{\restriction X}$ is continuous on A and $\inf A = \inf B$ and $\sup B = \inf C$ and $\sup C = \sup A$. Then $B \subseteq A$ and $C \subseteq A$ and $\int\limits_A f'_{\restriction X}(x)dx = \int\limits_B f'_{\restriction X}(x)dx + \int\limits_C f'_{\restriction X}(x)dx$.

Let a, b be elements of $\mathbb{R}$. Let us assume that $a \leqslant b$. The functor $['a, b']$ yields a closed-interval subset of $\mathbb{R}$ and is defined as follows:

(Def. 4) $['a, b'] = [a, b]$.

Let a, b be elements of $\mathbb{R}$ and let f be a partial function from $\mathbb{R}$ to $\mathbb{R}$. The functor $\int\limits_a^b f(x)dx$ yields a real number and is defined by:

(Def. 5) $$\int\limits_a^b f(x)dx = \begin{cases} \int\limits_{['a,b']} f(x)dx, & \text{if } a \leqslant b, \\ -\int\limits_{['b,a']} f(x)dx, & \text{otherwise.} \end{cases}$$

We now state three propositions:

(19) Let f be a partial function from $\mathbb{R}$ to $\mathbb{R}$, A be a closed-interval subset of $\mathbb{R}$, and given a, b. If $A = [a, b]$, then $\int\limits_A f(x)dx = \int\limits_a^b f(x)dx$.

(20) Let f be a partial function from $\mathbb{R}$ to $\mathbb{R}$, A be a closed-interval subset of

$\mathbb{R}$, and given a, b. If $A = [b, a]$, then $-\int\limits_{A} f(x)dx = \int\limits_{a}^{b} f(x)dx$.

(21) Let f, g be partial functions from $\mathbb{R}$ to $\mathbb{R}$ and X be an open subset of $\mathbb{R}$. Suppose that f is differentiable on X and g is differentiable on X and $A \subseteq X$ and $f'_{\restriction X}$ is integrable on A and $f'_{\restriction X}$ is bounded on A and $g'_{\restriction X}$ is integrable on A and $g'_{\restriction X}$ is bounded on A. Then $\int\limits_{A} f'_{\restriction X}\, g(x)dx = f(\sup A) \cdot g(\sup A) - f(\inf A) \cdot g(\inf A) - \int\limits_{A} f\, g'_{\restriction X}(x)dx$.

References

[1] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[2] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[3] Czesław Byliński. Partial functions. *Formalized Mathematics*, 1(**2**):357–367, 1990.

[4] Czesław Byliński. The sum and product of finite sequences of real numbers. *Formalized Mathematics*, 1(**4**):661–668, 1990.

[5] Czesław Byliński and Piotr Rudnicki. Bounding boxes for compact sets in $\mathcal{E}^2$. *Formalized Mathematics*, 6(**3**):427–440, 1997.

[6] Noboru Endou and Artur Korniłowicz. The definition of the Riemann definite integral and some related lemmas. *Formalized Mathematics*, 8(**1**):93–102, 1999.

[7] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[8] Jarosław Kotowicz. Convergent real sequences. Upper and lower bound of sets of real numbers. *Formalized Mathematics*, 1(**3**):477–481, 1990.

[9] Jarosław Kotowicz. Partial functions from a domain to the set of real numbers. *Formalized Mathematics*, 1(**4**):703–709, 1990.

[10] Jarosław Kotowicz. Properties of real functions. *Formalized Mathematics*, 1(**4**):781–786, 1990.

[11] Jarosław Kotowicz. Real sequences and basic operations on them. *Formalized Mathematics*, 1(**2**):269–272, 1990.

[12] Konrad Raczkowski and Paweł Sadowski. Real function continuity. *Formalized Mathematics*, 1(**4**):787–791, 1990.

[13] Konrad Raczkowski and Paweł Sadowski. Real function differentiability. *Formalized Mathematics*, 1(**4**):797–801, 1990.

[14] Konrad Raczkowski and Paweł Sadowski. Topological properties of subsets in real numbers. *Formalized Mathematics*, 1(**4**):777–780, 1990.

[15] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[16] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

Received March 23, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Introduction to Several Concepts of Convexity and Semicontinuity for Function from $\mathbb{R}$ to $\mathbb{R}$

Noboru Endou
Shinshu University
Nagano

Katsumi Wasaki
Shinshu University
Nagano

Yasunari Shidama
Shinshu University
Nagano

Summary. This article is an introduction to convex analysis. In the beginning, we have defined the concept of strictly convexity and proved some basic properties between convexity and strictly convexity. Moreover, we have defined concepts of other convexity and semicontinuity, and proved their basic properties.

MML Identifier: RFUNCT_4.

The papers [12], [3], [1], [4], [5], [9], [6], [13], [8], [16], [17], [11], [7], [10], [14], [15], and [2] provide the notation and terminology for this paper.

1. Some Useful Properties of n-Tuples on $\mathbb{R}$

We adopt the following convention: a, b, r, s, x_0, x are real numbers, f, g are partial functions from $\mathbb{R}$ to $\mathbb{R}$, and X, Y are sets.

The following propositions are true:

(1) $\max(a, b) \geqslant \min(a, b)$.

(2) Let n be a natural number, R_1, R_2 be elements of $\mathbb{R}^n$, and r_1, r_2 be real numbers. Then $R_1 \frown \langle r_1 \rangle \bullet R_2 \frown \langle r_2 \rangle = (R_1 \bullet R_2) \frown \langle r_1 \cdot r_2 \rangle$.

(3) Let n be a natural number and R be an element of $\mathbb{R}^n$. Suppose $\sum R = 0$ and for every natural number i such that $i \in \operatorname{dom} R$ holds $0 \leqslant R(i)$. Let i be a natural number. If $i \in \operatorname{dom} R$, then $R(i) = 0$.

ISSN 1426–2630

(4) Let n be a natural number and R be an element of $\mathbb{R}^n$. Suppose that for every natural number i such that $i \in \operatorname{dom} R$ holds $0 = R(i)$. Then $R = n \mapsto (0$ **qua** real number$)$.

(5) For every natural number n and for every element R of $\mathbb{R}^n$ holds $n \mapsto (0$ **qua** real number$) \bullet R = n \mapsto (0$ **qua** real number$)$.

2. Convex and Strictly Convex Functions

Let us consider f, X. We say that f is strictly convex on X if and only if the conditions (Def. 1) are satisfied.

(Def. 1)(i) $X \subseteq \operatorname{dom} f$, and

(ii) for every real number p such that $0 < p$ and $p < 1$ and for all real numbers r, s such that $r \in X$ and $s \in X$ and $p \cdot r + (1-p) \cdot s \in X$ and $r \neq s$ holds $f(p \cdot r + (1-p) \cdot s) < p \cdot f(r) + (1-p) \cdot f(s)$.

We now state a number of propositions:

(6) If f is strictly convex on X, then f is convex on X.

(7) Let a, b be real numbers and f be a partial function from $\mathbb{R}$ to $\mathbb{R}$. Then f is strictly convex on $[a,b]$ if and only if the following conditions are satisfied:

(i) $[a,b] \subseteq \operatorname{dom} f$, and

(ii) for every real number p such that $0 < p$ and $p < 1$ and for all real numbers r, s such that $r \in [a,b]$ and $s \in [a,b]$ and $r \neq s$ holds $f(p \cdot r + (1-p) \cdot s) < p \cdot f(r) + (1-p) \cdot f(s)$.

(8) Let X be a set and f be a partial function from $\mathbb{R}$ to $\mathbb{R}$. Then f is convex on X if and only if the following conditions are satisfied:

(i) $X \subseteq \operatorname{dom} f$, and

(ii) for all real numbers a, b, c such that $a \in X$ and $b \in X$ and $c \in X$ and $a < b$ and $b < c$ holds $f(b) \leqslant \frac{c-b}{c-a} \cdot f(a) + \frac{b-a}{c-a} \cdot f(c)$.

(9) Let X be a set and f be a partial function from $\mathbb{R}$ to $\mathbb{R}$. Then f is strictly convex on X if and only if the following conditions are satisfied:

(i) $X \subseteq \operatorname{dom} f$, and

(ii) for all real numbers a, b, c such that $a \in X$ and $b \in X$ and $c \in X$ and $a < b$ and $b < c$ holds $f(b) < \frac{c-b}{c-a} \cdot f(a) + \frac{b-a}{c-a} \cdot f(c)$.

(10) If f is strictly convex on X and $Y \subseteq X$, then f is strictly convex on Y.

(11) f is strictly convex on X iff $f - r$ is strictly convex on X.

(12) If $0 < r$, then f is strictly convex on X iff $r\,f$ is strictly convex on X.

(13) If f is strictly convex on X and g is strictly convex on X, then $f + g$ is strictly convex on X.

(14) If f is strictly convex on X and g is convex on X, then $f + g$ is strictly convex on X.

(15) Suppose f is strictly convex on X but g is strictly convex on X but $a > 0$ and $b \geqslant 0$ or $a \geqslant 0$ and $b > 0$. Then $a\,f + b\,g$ is strictly convex on X.

(16) f is convex on X if and only if the following conditions are satisfied:

(i) $X \subseteq \operatorname{dom} f$, and

(ii) for all a, b, r such that $a \in X$ and $b \in X$ and $r \in X$ and $a < r$ and $r < b$ holds $\frac{f(r)-f(a)}{r-a} \leqslant \frac{f(b)-f(a)}{b-a}$ and $\frac{f(b)-f(a)}{b-a} \leqslant \frac{f(b)-f(r)}{b-r}$.

(17) f is strictly convex on X if and only if the following conditions are satisfied:

(i) $X \subseteq \operatorname{dom} f$, and

(ii) for all a, b, r such that $a \in X$ and $b \in X$ and $r \in X$ and $a < r$ and $r < b$ holds $\frac{f(r)-f(a)}{r-a} < \frac{f(b)-f(a)}{b-a}$ and $\frac{f(b)-f(a)}{b-a} < \frac{f(b)-f(r)}{b-r}$.

(18) Let f be a partial function from $\mathbb{R}$ to $\mathbb{R}$. Suppose f is total. Then for every natural number n and for all elements P, E, F of $\mathbb{R}^n$ such that $\sum P = 1$ and for every natural number i such that $i \in \operatorname{dom} P$ holds $P(i) \geqslant 0$ and $F(i) = f(E(i))$ holds $f(\sum(P \bullet E)) \leqslant \sum(P \bullet F)$ if and only if f is convex on $\mathbb{R}$.

(19) Let f be a partial function from $\mathbb{R}$ to $\mathbb{R}$, I be an interval, and a be a real number. Suppose there exist real numbers x_1, x_2 such that $x_1 \in I$ and $x_2 \in I$ and $x_1 < a$ and $a < x_2$ and f is convex on I. Then f is continuous in a.

3. Definitions of Several Convexity and Semicontinuity Concepts

Let us consider f, X. We say that f is quasiconvex on X if and only if the conditions (Def. 2) are satisfied.

(Def. 2)(i) $X \subseteq \operatorname{dom} f$, and

(ii) for every real number p such that $0 < p$ and $p < 1$ and for all real numbers r, s such that $r \in X$ and $s \in X$ and $p \cdot r + (1-p) \cdot s \in X$ holds $f(p \cdot r + (1-p) \cdot s) \leqslant \max(f(r), f(s))$.

Let us consider f, X. We say that f is strictly quasiconvex on X if and only if the conditions (Def. 3) are satisfied.

(Def. 3)(i) $X \subseteq \operatorname{dom} f$, and

(ii) for every real number p such that $0 < p$ and $p < 1$ and for all real numbers r, s such that $r \in X$ and $s \in X$ and $p \cdot r + (1-p) \cdot s \in X$ and $f(r) \neq f(s)$ holds $f(p \cdot r + (1-p) \cdot s) < \max(f(r), f(s))$.

Let us consider f, X. We say that f is strongly quasiconvex on X if and only if the conditions (Def. 4) are satisfied.

(Def. 4)(i) $X \subseteq \operatorname{dom} f$, and

(ii) for every real number p such that $0 < p$ and $p < 1$ and for all real numbers r, s such that $r \in X$ and $s \in X$ and $p \cdot r + (1 - p) \cdot s \in X$ and $r \neq s$ holds $f(p \cdot r + (1 - p) \cdot s) < \max(f(r), f(s))$.

Let us consider f, x_0. We say that f is upper semicontinuous in x_0 if and only if:

(Def. 5) $x_0 \in \operatorname{dom} f$ and for every r such that $0 < r$ there exists s such that $0 < s$ and for every x such that $x \in \operatorname{dom} f$ and $|x - x_0| < s$ holds $f(x_0) - f(x) < r$.

Let us consider f, X. We say that f is upper semicontinuous on X if and only if:

(Def. 6) $X \subseteq \operatorname{dom} f$ and for every x_0 such that $x_0 \in X$ holds $f \restriction X$ is upper semicontinuous in x_0.

Let us consider f, x_0. We say that f is lower semicontinuous in x_0 if and only if:

(Def. 7) $x_0 \in \operatorname{dom} f$ and for every r such that $0 < r$ there exists s such that $0 < s$ and for every x such that $x \in \operatorname{dom} f$ and $|x - x_0| < s$ holds $f(x) - f(x_0) < r$.

Let us consider f, X. We say that f is lower semicontinuous on X if and only if:

(Def. 8) $X \subseteq \operatorname{dom} f$ and for every x_0 such that $x_0 \in X$ holds $f \restriction X$ is lower semicontinuous in x_0.

The following propositions are true:

(20) Let given x_0, f. Then f is upper semicontinuous in x_0 and f is lower semicontinuous in x_0 if and only if f is continuous in x_0.

(21) Let given X, f. Then f is upper semicontinuous on X and f is lower semicontinuous on X if and only if f is continuous on X.

(22) For all X, f such that f is strictly convex on X holds f is strongly quasiconvex on X.

(23) For all X, f such that f is strongly quasiconvex on X holds f is quasiconvex on X.

(24) For all X, f such that f is convex on X holds f is strictly quasiconvex on X.

(25) For all X, f such that f is strongly quasiconvex on X holds f is strictly quasiconvex on X.

(26) Let given X, f. Suppose f is strictly quasiconvex on X and f is one-to-one. Then f is strongly quasiconvex on X.

References

[1] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[2] Józef Białas. Infimum and supremum of the set of real numbers. Measure theory. *Formalized Mathematics*, 2(**1**):163–171, 1991.

[3] Czesław Byliński. Binary operations applied to finite sequences. *Formalized Mathematics*, 1(**4**):643–649, 1990.

[4] Czesław Byliński. Finite sequences and tuples of elements of a non-empty sets. *Formalized Mathematics*, 1(**3**):529–536, 1990.

[5] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[6] Czesław Byliński. Partial functions. *Formalized Mathematics*, 1(**2**):357–367, 1990.

[7] Czesław Byliński. The sum and product of finite sequences of real numbers. *Formalized Mathematics*, 1(**4**):661–668, 1990.

[8] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[9] Białas Józef. Properties of the intervals of real numbers. *Formalized Mathematics*, 3(**2**):263–269, 1992.

[10] Jarosław Kotowicz. Real sequences and basic operations on them. *Formalized Mathematics*, 1(**2**):269–272, 1990.

[11] Jarosław Kotowicz and Yuji Sakai. Properties of partial functions from a domain to the set of real numbers. *Formalized Mathematics*, 3(**2**):279–288, 1992.

[12] Konrad Raczkowski and Paweł Sadowski. Real function continuity. *Formalized Mathematics*, 1(**4**):787–791, 1990.

[13] Konrad Raczkowski and Paweł Sadowski. Topological properties of subsets in real numbers. *Formalized Mathematics*, 1(**4**):777–780, 1990.

[14] Andrzej Trybulec and Czesław Byliński. Some properties of real numbers. *Formalized Mathematics*, 1(**3**):445–449, 1990.

[15] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[16] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

[17] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

Received March 23, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Standard Ordering of Instruction Locations

Andrzej Trybulec
University of Białystok

Piotr Rudnicki
University of Alberta

Artur Korniłowicz
University of Białystok

MML Identifier: AMISTD_1.

The notation and terminology used in this paper have been introduced in the following articles: [11], [15], [12], [18], [1], [3], [14], [4], [16], [6], [7], [8], [9], [2], [10], [5], [19], [20], [13], and [17].

1. Preliminaries

We use the following convention: x, X are sets, D is a non empty set, and k, m, n are natural numbers.

The following two propositions are true:

(1) For every real number r holds $\max\{r\} = r$.

(2) $\max\{n\} = n$.

One can verify that there exists a finite sequence which is non trivial.

The following proposition is true

(3) For every trivial finite sequence f of elements of D holds f is empty or there exists an element x of D such that $f = \langle x \rangle$.

Let x, y be sets. Note that $\langle x, y \rangle$ is non empty.

Let us observe that every binary relation has non empty elements.

One can prove the following proposition

(4) id_X is bijective.

Let A be a finite set and let B be a set. Observe that $A \longmapsto B$ is finite.

Let x, y be sets. One can check that $x \dot{\longmapsto} y$ is trivial.

2. Restricted Concatenation

Let f_1 be a non empty finite sequence and let f_2 be a finite sequence. Observe that $f_1 \frown f_2$ is non empty.

The following propositions are true:

(5) Let f_1 be a non empty finite sequence of elements of D and f_2 be a finite sequence of elements of D. Then $(f_1 \frown f_2)_1 = (f_1)_1$.

(6) Let f_1 be a finite sequence of elements of D and f_2 be a non trivial finite sequence of elements of D. Then $(f_1 \frown f_2)_{\mathrm{len}(f_1 \frown f_2)} = (f_2)_{\mathrm{len}\, f_2}$.

(7) For every finite sequence f holds $f \frown \varepsilon = f$.

(8) For every finite sequence f holds $f \frown \langle x \rangle = f$.

(9) For all finite sequences f_1, f_2 of elements of D such that $1 \leqslant n$ and $n \leqslant \mathrm{len}\, f_1$ holds $(f_1 \frown f_2)_n = (f_1)_n$.

(10) For all finite sequences f_1, f_2 of elements of D such that $1 \leqslant n$ and $n < \mathrm{len}\, f_2$ holds $(f_1 \frown f_2)_{\mathrm{len}\, f_1 + n} = (f_2)_{n+1}$.

3. Ami-Struct

For simplicity, we adopt the following convention: N is a set with non empty elements, S is a von Neumann definite AMI over N, i is an instruction of S, l, l_1, l_2, l_3 are instruction-locations of S, and s is a state of S.

We now state the proposition

(11) Let S be a definite AMI over N, I be an instruction of S, and s be a state of S. Then $s+\cdot(($the instruction locations of $S) \longmapsto I)$ is a state of S.

Let N be a set and let S be an AMI over N. Observe that every finite partial state of S which is empty is also programmed.

Let N be a set and let S be an AMI over N. One can check that there exists a finite partial state of S which is empty.

Let N be a set with non empty elements and let S be a von Neumann definite AMI over N. Note that there exists a finite partial state of S which is non empty, trivial, and programmed.

Let N be a set with non empty elements, let S be an AMI over N, let i be an instruction of S, and let s be a state of S. One can verify that (the execution of $S)(i)(s)$ is function-like and relation-like.

Let N be a set and let S be an AMI over N.

(Def. 1) An element of the instruction codes of S is said to be an instruction type of S.

Let N be a set, let S be an AMI over N, and let I be an element of the instructions of S. The functor $\mathrm{InsCode}(I)$ yields an instruction type of S and is defined by:

(Def. 2) $\mathrm{InsCode}(I) = I_{\mathbf{1}}$.

Let N be a set with non empty elements and let S be a steady-programmed von Neumann definite AMI over N. Observe that there exists a finite partial state of S which is non empty, trivial, autonomic, and programmed.

One can prove the following propositions:

(12) Let S be a steady-programmed von Neumann definite AMI over N, i_1 be an instruction-location of S, and I be an instruction of S. Then $i_1 \overset{\cdot}{\longmapsto} I$ is autonomic.

(13) Every steady-programmed von Neumann definite AMI over N is programmable.

Let N be a set with non empty elements. One can check that every von Neumann definite AMI over N which is steady-programmed is also programmable.

Let N be a set with non empty elements, let S be an AMI over N, and let T be an instruction type of S. We say that T is jump-only if and only if the condition (Def. 3) is satisfied.

(Def. 3) Let s be a state of S, o be an object of S, and I be an instruction of S. If $\mathrm{InsCode}(I) = T$ and $o \neq \mathbf{IC}_S$, then $(\mathrm{Exec}(I, s))(o) = s(o)$.

Let N be a set with non empty elements, let S be an AMI over N, and let I be an instruction of S. We say that I is jump-only if and only if:

(Def. 4) $\mathrm{InsCode}(I)$ is jump-only.

Let us consider N, S, i, l. The functor $\mathrm{NIC}(i, l)$ yielding a subset of the instruction locations of S is defined by:

(Def. 5) $\mathrm{NIC}(i, l) = \{\mathbf{IC}_{\mathrm{Following}(s)} : \mathbf{IC}_s = l \ \wedge \ s(l) = i\}$.

Let N be a set with non empty elements, let S be a realistic von Neumann definite AMI over N, let i be an instruction of S, and let l be an instruction-location of S. Note that $\mathrm{NIC}(i, l)$ is non empty.

Let us consider N, S, i. The functor $\mathrm{JUMP}(i)$ yields a subset of the instruction locations of S and is defined by:

(Def. 6) $\mathrm{JUMP}(i) = \bigcap\{\mathrm{NIC}(i, l)\}$.

Let us consider N, S, l. The functor $\mathrm{SUCC}(l)$ yielding a subset of the instruction locations of S is defined by:

(Def. 7) $\mathrm{SUCC}(l) = \bigcup\{\mathrm{NIC}(i, l) \setminus \mathrm{JUMP}(i)\}$.

One can prove the following propositions:

(14) Let S be a von Neumann definite AMI over N and i be an instruction of S. Suppose the instruction locations of S are non trivial and for every instruction-location l of S holds $\mathrm{NIC}(i, l) = \{l\}$. Then $\mathrm{JUMP}(i)$ is empty.

(15) Let S be a realistic von Neumann definite AMI over N, i_1 be an instruction-location of S, and i be an instruction of S. If i is halting, then $\mathrm{NIC}(i, i_1) = \{i_1\}$.

4. Ordering of Instruction Locations

Let us consider N, S, l_1, l_2. The predicate $l_1 \leqslant l_2$ is defined by the condition (Def. 8).

(Def. 8) There exists a non empty finite sequence f of elements of the instruction locations of S such that $f_1 = l_1$ and $f_{\mathrm{len}\, f} = l_2$ and for every n such that $1 \leqslant n$ and $n < \mathrm{len}\, f$ holds $f_{n+1} \in \mathrm{SUCC}(f_n)$.

Let us note that the predicate $l_1 \leqslant l_2$ is reflexive.

Next we state the proposition

(16) If $l_1 \leqslant l_2$ and $l_2 \leqslant l_3$, then $l_1 \leqslant l_3$.

Let us consider N, S. We say that S is InsLoc-antisymmetric if and only if:

(Def. 9) For all l_1, l_2 such that $l_1 \leqslant l_2$ and $l_2 \leqslant l_1$ holds $l_1 = l_2$.

Let us consider N, S. We say that S is standard if and only if the condition (Def. 10) is satisfied.

(Def. 10) There exists a function f from $\mathbb{N}$ into the instruction locations of S such that f is bijective and for all natural numbers m, n holds $m \leqslant n$ iff $f(m) \leqslant f(n)$.

One can prove the following three propositions:

(17) Let S be a von Neumann definite AMI over N and f_1, f_2 be functions from $\mathbb{N}$ into the instruction locations of S. Suppose that

(i) f_1 is bijective,

(ii) for all natural numbers m, n holds $m \leqslant n$ iff $f_1(m) \leqslant f_1(n)$,

(iii) f_2 is bijective, and

(iv) for all natural numbers m, n holds $m \leqslant n$ iff $f_2(m) \leqslant f_2(n)$.

Then $f_1 = f_2$.

(18) Let S be a von Neumann definite AMI over N and f be a function from $\mathbb{N}$ into the instruction locations of S. Suppose f is bijective. Then the following statements are equivalent

(i) for all natural numbers m, n holds $m \leqslant n$ iff $f(m) \leqslant f(n)$,

(ii) for every natural number k holds $f(k+1) \in \mathrm{SUCC}(f(k))$ and for every natural number j such that $f(j) \in \mathrm{SUCC}(f(k))$ holds $k \leqslant j$.

(19) Let S be a von Neumann definite AMI over N. Then S is standard if and only if there exists a function f from $\mathbb{N}$ into the instruction locations of S such that f is bijective and for every natural number k holds $f(k+1) \in$

SUCC($f(k)$) and for every natural number j such that $f(j) \in$ SUCC($f(k)$) holds $k \leqslant j$.

5. Standard Trivial Computer

Let N be a set with non empty elements. The functor STC(N) yielding a strict AMI over N is defined by the conditions (Def. 11).

(Def. 11) The objects of STC(N) $= \mathbb{N} \cup \{\mathbb{N}\}$ and the instruction counter of STC(N) $= \mathbb{N}$ and the instruction locations of STC(N) $= \mathbb{N}$ and the instruction codes of STC(N) $= \{0, 1\}$ and the instructions of STC(N) $= \{\langle 0,\ 0\rangle, \langle 1,\ 0\rangle\}$ and the object kind of STC(N) $= (\mathbb{N} \longmapsto \{\langle 1,\ 0\rangle, \langle 0,\ 0\rangle\}) {+}{\cdot} (\{\mathbb{N}\} \longmapsto \mathbb{N})$ and there exists a function f from $\prod$(the object kind of STC(N)) into $\prod$(the object kind of STC(N)) such that for every element s of $\prod$(the object kind of STC(N)) holds $f(s) = s {+}{\cdot} (\{\mathbb{N}\} \longmapsto \operatorname{succ} s(\mathbb{N}))$ and the execution of STC(N) $= (\{\langle 1,\ 0\rangle\} \longmapsto f) {+}{\cdot} (\{\langle 0,\ 0\rangle\} \longmapsto \mathrm{id}_{\prod (\text{the object kind of STC}(N))})$.

Let N be a set with non empty elements. Note that the instruction locations of STC(N) is infinite.

Let N be a set with non empty elements. Observe that STC(N) is von Neumann definite realistic steady-programmed and data-oriented.

Next we state several propositions:

(20) For every instruction i of STC(N) such that InsCode(i) $= 0$ holds i is halting.

(21) For every instruction i of STC(N) such that InsCode(i) $= 1$ holds i is non halting.

(22) For every instruction i of STC(N) holds InsCode(i) $= 1$ or InsCode(i) $= 0$.

(23) Every instruction of STC(N) is jump-only.

(24) For every instruction-location l of STC(N) such that $l = k$ holds SUCC(l) $= \{k, k+1\}$.

Let N be a set with non empty elements. Observe that STC(N) is standard.

Let N be a set with non empty elements. Observe that STC(N) is halting.

Let N be a set with non empty elements. One can check that there exists a von Neumann definite AMI over N which is standard, halting, realistic, steady-programmed, and programmable.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, and let k be a natural number. The functor $\mathrm{il}_S(k)$ yields an instruction-location of S and is defined by the condition (Def. 12).

(Def. 12) There exists a function f from $\mathbb{N}$ into the instruction locations of S such that f is bijective and for all natural numbers m, n holds $m \leqslant n$ iff $f(m) \leqslant f(n)$ and $\mathrm{il}_S(k) = f(k)$.

We now state two propositions:

(25) Let S be a standard von Neumann definite AMI over N and k_1, k_2 be natural numbers. If $\mathrm{il}_S(k_1) = \mathrm{il}_S(k_2)$, then $k_1 = k_2$.

(26) Let S be a standard von Neumann definite AMI over N and l be an instruction-location of S. Then there exists a natural number k such that $l = \mathrm{il}_S(k)$.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, and let l be an instruction-location of S. The functor $\mathrm{locnum}(l)$ yields a natural number and is defined as follows:

(Def. 13) $\mathrm{il}_S(\mathrm{locnum}(l)) = l$.

One can prove the following propositions:

(27) Let S be a standard von Neumann definite AMI over N and l_1, l_2 be instruction-locations of S. If $\mathrm{locnum}(l_1) = \mathrm{locnum}(l_2)$, then $l_1 = l_2$.

(28) Let S be a standard von Neumann definite AMI over N and k_1, k_2 be natural numbers. Then $\mathrm{il}_S(k_1) \leqslant \mathrm{il}_S(k_2)$ if and only if $k_1 \leqslant k_2$.

(29) Let S be a standard von Neumann definite AMI over N and l_1, l_2 be instruction-locations of S. Then $\mathrm{locnum}(l_1) \leqslant \mathrm{locnum}(l_2)$ if and only if $l_1 \leqslant l_2$.

(30) If S is standard, then S is InsLoc-antisymmetric.

Let us consider N. Observe that every von Neumann definite AMI over N which is standard is also InsLoc-antisymmetric.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, let f be an instruction-location of S, and let k be a natural number. The functor $f + k$ yielding an instruction-location of S is defined by:

(Def. 14) $f + k = \mathrm{il}_S(\mathrm{locnum}(f) + k)$.

Next we state three propositions:

(31) For every standard von Neumann definite AMI S over N and for every instruction-location f of S holds $f + 0 = f$.

(32) Let S be a standard von Neumann definite AMI over N and f, g be instruction-locations of S. If $f + k = g + k$, then $f = g$.

(33) For every standard von Neumann definite AMI S over N and for every instruction-location f of S holds $\mathrm{locnum}(f) + k = \mathrm{locnum}(f + k)$.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, and let f be an instruction-location of S. The functor $\mathrm{NextLoc}\, f$ yields an instruction-location of S and is defined as follows:

(Def. 15) $\mathrm{NextLoc}\, f = f + 1$.

The following propositions are true:

(34) For every standard von Neumann definite AMI S over N and for every instruction-location f of S holds $\mathrm{NextLoc}\, f = \mathrm{il}_S(\mathrm{locnum}(f)+1)$.

(35) For every standard von Neumann definite AMI S over N and for every instruction-location f of S holds $f \neq \mathrm{NextLoc}\, f$.

(36) Let S be a standard von Neumann definite AMI over N and f, g be instruction-locations of S. If $\mathrm{NextLoc}\, f = \mathrm{NextLoc}\, g$, then $f = g$.

(37) $\mathrm{il}_{\mathrm{STC}(N)}(k) = k$.

(38) For every instruction i of $\mathrm{STC}(N)$ and for every state s of $\mathrm{STC}(N)$ such that $\mathrm{InsCode}(i) = 1$ holds $(\mathrm{Exec}(i,s))(\mathbf{IC}_{\mathrm{STC}(N)}) = \mathrm{NextLoc}\, \mathbf{IC}_s$.

(39) For every instruction-location l of $\mathrm{STC}(N)$ and for every instruction i of $\mathrm{STC}(N)$ such that $\mathrm{InsCode}(i) = 1$ holds $\mathrm{NIC}(i,l) = \{\mathrm{NextLoc}\, l\}$.

(40) For every instruction-location l of $\mathrm{STC}(N)$ holds $\mathrm{SUCC}(l) = \{l, \mathrm{NextLoc}\, l\}$.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, and let i be an instruction of S. We say that i is sequential if and only if:

(Def. 16) For every state s of S holds $(\mathrm{Exec}(i,s))(\mathbf{IC}_S) = \mathrm{NextLoc}\, \mathbf{IC}_s$.

The following propositions are true:

(41) Let S be a standard realistic von Neumann definite AMI over N, i_1 be an instruction-location of S, and i be an instruction of S. If i is sequential, then $\mathrm{NIC}(i, i_1) = \{\mathrm{NextLoc}\, i_1\}$.

(42) Let S be a realistic standard von Neumann definite AMI over N and i be an instruction of S. If i is sequential, then i is non halting.

Let us consider N and let S be a realistic standard von Neumann definite AMI over N. Observe that every instruction of S which is sequential is also non halting and every instruction of S which is halting is also non sequential.

One can prove the following proposition

(43) Let S be a standard von Neumann definite AMI over N and i be an instruction of S. If $\mathrm{JUMP}(i)$ is non empty, then i is non sequential.

6. Closedness of Finite Partial States

Let N be a set with non empty elements, let S be a von Neumann definite AMI over N, and let F be a finite partial state of S. We say that F is closed if and only if:

(Def. 17) For every instruction-location l of S such that $l \in \mathrm{dom}\, F$ holds $\mathrm{NIC}(\pi_l F, l) \subseteq \mathrm{dom}\, F$.

We say that F is really-closed if and only if:

(Def. 18) For every state s of S such that $F \subseteq s$ and $\mathbf{IC}_s \in \operatorname{dom} F$ and for every natural number k holds $\mathbf{IC}_{(\mathrm{Computation}(s))(k)} \in \operatorname{dom} F$.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, and let F be a finite partial state of S. We say that F is para-closed if and only if:

(Def. 19) For every state s of S such that $F \subseteq s$ and $\mathbf{IC}_s = \mathrm{il}_S(0)$ and for every natural number k holds $\mathbf{IC}_{(\mathrm{Computation}(s))(k)} \in \operatorname{dom} F$.

The following propositions are true:

(44) Let S be a standard steady-programmed von Neumann definite AMI over N and F be a finite partial state of S. If F is really-closed and $\mathrm{il}_S(0) \in \operatorname{dom} F$, then F is para-closed.

(45) Let S be a von Neumann definite steady-programmed AMI over N and F be a finite partial state of S. If F is closed, then F is really-closed.

Let N be a set with non empty elements and let S be a von Neumann definite steady-programmed AMI over N. One can verify that every finite partial state of S which is closed is also really-closed.

We now state the proposition

(46) For every standard realistic halting von Neumann definite AMI S over N holds $\mathrm{il}_S(0) \dot{\longmapsto} \mathbf{halt}_S$ is closed.

Let N be a set with non empty elements, let S be a von Neumann definite AMI over N, and let F be a finite partial state of S. We say that F is lower if and only if the condition (Def. 20) is satisfied.

(Def. 20) Let l be an instruction-location of S. Suppose $l \in \operatorname{dom} F$. Let m be an instruction-location of S. If $m \leqslant l$, then $m \in \operatorname{dom} F$.

The following proposition is true

(47) For every von Neumann definite AMI S over N holds every empty finite partial state of S is lower.

Let N be a set with non empty elements and let S be a von Neumann definite AMI over N. Observe that every finite partial state of S which is empty is also lower.

The following proposition is true

(48) For every standard von Neumann definite AMI S over N and for every instruction i of S holds $\mathrm{il}_S(0) \dot{\longmapsto} i$ is lower.

Let N be a set with non empty elements and let S be a standard von Neumann definite AMI over N. Note that there exists a finite partial state of S which is lower, non empty, trivial, and programmed.

We now state two propositions:

(49) Let S be a standard von Neumann definite AMI over N and F be a lower non empty programmed finite partial state of S. Then $\mathrm{il}_S(0) \in \mathrm{dom}\,F$.

(50) Let N be a set with non empty elements, S be a standard von Neumann definite AMI over N, and P be a lower programmed finite partial state of S. Then $m < \mathrm{card}\,P$ if and only if $\mathrm{il}_S(m) \in \mathrm{dom}\,P$.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, and let F be a non empty programmed finite partial state of S. The functor $\mathrm{LastLoc}\,F$ yields an instruction-location of S and is defined by the condition (Def. 21).

(Def. 21) There exists a finite non empty subset M of $\mathbb{N}$ such that $M = \{\mathrm{locnum}(l); l$ ranges over elements of the instruction locations of S: $l \in \mathrm{dom}\,F\}$ and $\mathrm{LastLoc}\,F = \mathrm{il}_S(\max M)$.

We now state several propositions:

(51) Let S be a standard von Neumann definite AMI over N and F be a non empty programmed finite partial state of S. Then $\mathrm{LastLoc}\,F \in \mathrm{dom}\,F$.

(52) Let S be a standard von Neumann definite AMI over N and F, G be non empty programmed finite partial states of S. If $F \subseteq G$, then $\mathrm{LastLoc}\,F \leqslant \mathrm{LastLoc}\,G$.

(53) Let S be a standard von Neumann definite AMI over N, F be a non empty programmed finite partial state of S, and l be an instruction-location of S. If $l \in \mathrm{dom}\,F$, then $l \leqslant \mathrm{LastLoc}\,F$.

(54) Let S be a standard von Neumann definite AMI over N, F be a lower non empty programmed finite partial state of S, and G be a non empty programmed finite partial state of S. If $F \subseteq G$ and $\mathrm{LastLoc}\,F = \mathrm{LastLoc}\,G$, then $F = G$.

(55) Let N be a set with non empty elements, S be a standard von Neumann definite AMI over N, and F be a lower non empty programmed finite partial state of S. Then $\mathrm{LastLoc}\,F = \mathrm{il}_S(\mathrm{card}\,F -' 1)$.

Let N be a set with non empty elements and let S be a standard steady-programmed von Neumann definite AMI over N. Note that every finite partial state of S which is really-closed, lower, non empty, and programmed is also para-closed.

Let N be a set with non empty elements, let S be a standard halting von Neumann definite AMI over N, and let F be a non empty programmed finite partial state of S. We say that F is halt-ending if and only if:

(Def. 22) $F(\mathrm{LastLoc}\,F) = \mathbf{halt}_S$.

We say that F is unique-halt if and only if:

(Def. 23) For every instruction-location f of S such that $F(f) = \mathbf{halt}_S$ and $f \in \mathrm{dom}\,F$ holds $f = \mathrm{LastLoc}\,F$.

Let N be a set with non empty elements and let S be a standard halting von Neumann definite AMI over N. One can check that there exists a lower non empty programmed finite partial state of S which is halt-ending, unique-halt, and trivial.

Let N be a set with non empty elements and let S be a standard halting realistic von Neumann definite AMI over N. One can check that there exists a finite partial state of S which is trivial, closed, lower, non empty, and programmed.

Let N be a set with non empty elements and let S be a standard halting realistic von Neumann definite AMI over N. Observe that there exists a lower non empty programmed finite partial state of S which is halt-ending, unique-halt, trivial, and closed.

Let N be a set with non empty elements and let S be a standard halting realistic steady-programmed von Neumann definite AMI over N. Observe that there exists a lower non empty programmed finite partial state of S which is halt-ending, unique-halt, autonomic, trivial, and closed.

Let N be a set with non empty elements and let S be a standard halting von Neumann definite AMI over N.

(Def. 24) A halt-ending unique-halt lower non empty programmed finite partial state of S is said to be a pre-Macro of S.

Let N be a set with non empty elements and let S be a standard realistic halting von Neumann definite AMI over N. One can verify that there exists a pre-Macro of S which is closed.

References

[1] Grzegorz Bancerek. Cardinal numbers. *Formalized Mathematics*, 1(**2**):377–382, 1990.

[2] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[3] Grzegorz Bancerek. König's theorem. *Formalized Mathematics*, 1(**3**):589–593, 1990.

[4] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[5] Józef Białas. Group and field definitions. *Formalized Mathematics*, 1(**3**):433–439, 1990.

[6] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[7] Czesław Byliński. Functions from a set to a set. *Formalized Mathematics*, 1(**1**):153–164, 1990.

[8] Czesław Byliński. The modification of a function by a function and the iteration of the composition of a function. *Formalized Mathematics*, 1(**3**):521–527, 1990.

[9] Yatsuka Nakamura and Piotr Rudnicki. Vertex sequences induced by chains. *Formalized Mathematics*, 5(**3**):297–304, 1996.

[10] Yatsuka Nakamura, Piotr Rudnicki, Andrzej Trybulec, and Pauline N. Kawamoto. Preliminaries to circuits, I. *Formalized Mathematics*, 5(**2**):167–172, 1996.

[11] Yatsuka Nakamura and Andrzej Trybulec. A mathematical model of CPU. *Formalized Mathematics*, 3(**2**):151–160, 1992.

[12] Takaya Nishiyama and Yasuho Mizuhara. Binary arithmetics. *Formalized Mathematics*, 4(**1**):83–86, 1993.

[13] Yozo Toda. The formalization of simple graphs. *Formalized Mathematics*, 5(**1**):137–144, 1996.

[14] Andrzej Trybulec. Domains and their Cartesian products. *Formalized Mathematics*, 1(**1**):115–122, 1990.

[15] Andrzej Trybulec and Yatsuka Nakamura. Some remarks on the simple concrete model of computer. *Formalized Mathematics*, 4(**1**):51–56, 1993.

[16] Wojciech A. Trybulec. Pigeon hole principle. *Formalized Mathematics*, 1(**3**):575–579, 1990.

[17] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[18] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

[19] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

[20] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

Received April 14, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

On the Composition of Macro Instructions of Standard Computers

Artur Korniłowicz
University of Białystok

MML Identifier: AMISTD_2.

The terminology and notation used in this paper are introduced in the following papers: [18], [11], [17], [12], [20], [1], [3], [14], [4], [8], [15], [5], [6], [2], [10], [9], [21], [13], [19], [16], and [7].

1. Preliminaries

We follow the rules: k, m are natural numbers, x, X are sets, and N is a set with non empty elements.

Let f be a function and let g be a non empty function. One can verify that $f{+\cdot}g$ is non empty and $g{+\cdot}f$ is non empty.

Let f, g be finite functions. Note that $f{+\cdot}g$ is finite.

Next we state two propositions:

(1) For all functions f, g holds $\operatorname{dom} f \approx \operatorname{dom} g$ iff $f \approx g$.

(2) For all finite functions f, g such that $\operatorname{dom} f \cap \operatorname{dom} g = \emptyset$ holds $\operatorname{card}(f{+\cdot}g) = \operatorname{card} f + \operatorname{card} g$.

Let f be a function and let A be a set. Note that $f \setminus A$ is function-like and relation-like.

One can prove the following two propositions:

(3) For all functions f, g such that $x \in \operatorname{dom} f \setminus \operatorname{dom} g$ holds $(f \setminus g)(x) = f(x)$.

(4) For every non empty finite set F holds $\operatorname{card} F - 1 = \operatorname{card} F -' 1$.

ISSN 1426–2630

2. Product Like Sets

Let S be a functional set. The functor $\prod_S$ yields a function and is defined as follows:

(Def. 1)(i) For every set x holds $x \in \operatorname{dom} \prod_S$ iff for every function f such that $f \in S$ holds $x \in \operatorname{dom} f$ and for every set i such that $i \in \operatorname{dom} \prod_S$ holds $\prod_S(i) = \pi_i S$ if S is non empty,

(ii) $\prod_S = \emptyset$, otherwise.

The following two propositions are true:

(5) For every non empty functional set S holds $\operatorname{dom} \prod_S = \bigcap \{\operatorname{dom} f : f$ ranges over elements of $S\}$.

(6) For every non empty functional set S and for every set i such that $i \in \operatorname{dom} \prod_S$ holds $\prod_S(i) = \{f(i) : f$ ranges over elements of $S\}$.

Let S be a set. We say that S is product-like if and only if:

(Def. 2) There exists a function f such that $S = \prod f$.

Let f be a function. One can check that $\prod f$ is product-like.

Let us mention that every set which is product-like is also functional and has common domain.

Let us observe that there exists a set which is product-like and non empty.

The following four propositions are true:

(7) For every functional set S with common domain holds $\operatorname{dom} \prod_S = \operatorname{DOM}(S)$.

(8) For every functional set S and for every set i such that $i \in \operatorname{dom} \prod_S$ holds $\prod_S(i) = \pi_i S$.

(9) For every functional set S with common domain holds $S \subseteq \prod \prod_S$.

(10) For every non empty product-like set S holds $S = \prod \prod_S$.

Let D be a set. Observe that every set of finite sequences of D is functional.

Let i be a natural number and let D be a set. One can check that D^i has common domain.

Let i be a natural number and let D be a set. Note that D^i is product-like.

3. Properties of AMI-Struct

One can prove the following propositions:

(11) Let N be a set, S be an AMI over N, and F be a finite partial state of S. Then $F \setminus X$ is a finite partial state of S.

(12) Let S be a von Neumann definite AMI over N and F be a programmed finite partial state of S. Then $F \setminus X$ is a programmed finite partial state of S.

Let N be a set with non empty elements, let S be a von Neumann definite AMI over N, let i_1, i_2 be instruction-locations of S, and let I_1, I_2 be elements of the instructions of S. Then $[i_1 \longmapsto I_1, i_2 \longmapsto I_2]$ is a finite partial state of S.

Let N be a set with non empty elements and let S be a halting AMI over N. Observe that there exists an instruction of S which is halting.

We now state three propositions:

(13) Let S be a standard von Neumann definite AMI over N, F be a lower programmed finite partial state of S, and G be a programmed finite partial state of S. If $\operatorname{dom} F = \operatorname{dom} G$, then G is lower.

(14) Let S be a standard von Neumann definite AMI over N, F be a lower programmed finite partial state of S, and f be an instruction-location of S. If $f \in \operatorname{dom} F$, then $\operatorname{locnum}(f) < \operatorname{card} F$.

(15) Let S be a standard von Neumann definite AMI over N and F be a lower programmed finite partial state of S. Then $\operatorname{dom} F = \{\mathrm{il}_S(k); k$ ranges over natural numbers: $k < \operatorname{card} F\}$.

Let N be a set, let S be an AMI over N, and let I be an element of the instructions of S. The functor AddressPart(I) is defined by:

(Def. 3) $\mathrm{AddressPart}(I) = I_{\mathbf{2}}$.

Let N be a set, let S be an AMI over N, and let I be an element of the instructions of S. Then AddressPart(I) is a finite sequence of elements of $\bigcup N \cup$ the objects of S.

We now state the proposition

(16) Let N be a set, S be an AMI over N, and I, J be elements of the instructions of S. If $\mathrm{InsCode}(I) = \mathrm{InsCode}(J)$ and $\mathrm{AddressPart}(I) = \mathrm{AddressPart}(J)$, then $I = J$.

Let N be a set and let S be an AMI over N. We say that S is homogeneous if and only if:

(Def. 4) For all instructions I, J of S such that $\mathrm{InsCode}(I) = \mathrm{InsCode}(J)$ holds $\operatorname{dom} \mathrm{AddressPart}(I) = \operatorname{dom} \mathrm{AddressPart}(J)$.

The following proposition is true

(17) For every instruction I of $\mathrm{STC}(N)$ holds $\mathrm{AddressPart}(I) = 0$.

Let N be a set, let S be an AMI over N, and let T be an instruction type of S. The functor AddressParts T is defined by:

(Def. 5) $\mathrm{AddressParts}\, T = \{\mathrm{AddressPart}(I); I$ ranges over instructions of S: $\mathrm{InsCode}(I) = T\}$.

Let N be a set, let S be an AMI over N, and let T be an instruction type of S. One can check that AddressParts T is functional.

Let N be a set with non empty elements, let S be a von Neumann definite AMI over N, and let I be an instruction of S. We say that I is explicit-jump-instruction if and only if the condition (Def. 6) is satisfied.

(Def. 6) Let f be a set. Suppose $f \in \mathrm{JUMP}(I)$. Then there exists a set k such that $k \in \mathrm{dom}\,\mathrm{AddressPart}(I)$ and $f = (\mathrm{AddressPart}(I))(k)$ and $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(I)}(k) =$ the instruction locations of S.

We say that I has ins-loc-in-jump if and only if the condition (Def. 7) is satisfied.

(Def. 7) Let f be a set. Given a set k such that $k \in \mathrm{dom}\,\mathrm{AddressPart}(I)$ and $f = (\mathrm{AddressPart}(I))(k)$ and $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(I)}(k) =$ the instruction locations of S. Then $f \in \mathrm{JUMP}(I)$.

Let N be a set with non empty elements and let S be a von Neumann definite AMI over N. We say that S is explicit-jump-instruction if and only if:

(Def. 8) Every instruction of S is explicit-jump-instruction.

We say that S has ins-loc-in-jump if and only if:

(Def. 9) Every instruction of S has ins-loc-in-jump.

Let N be a set and let S be an AMI over N. We say that S has non trivial instruction locations if and only if:

(Def. 10) The instruction locations of S are non trivial.

Let N be a set with non empty elements. Note that every von Neumann definite AMI over N which is standard has non trivial instruction locations.

Let N be a set with non empty elements. One can verify that there exists a von Neumann definite AMI over N which is standard.

Let N be a set with non empty elements and let S be an AMI over N with non trivial instruction locations. Observe that the instruction locations of S is non trivial.

The following proposition is true

(18) Let S be a standard von Neumann definite AMI over N and I be an instruction of S. If for every instruction-location f of S holds $\mathrm{NIC}(I, f) = \{\mathrm{NextLoc}\,f\}$, then $\mathrm{JUMP}(I)$ is empty.

Let N be a set with non empty elements and let I be an instruction of $\mathrm{STC}(N)$. Observe that $\mathrm{JUMP}(I)$ is empty.

Let N be a set and let S be an AMI over N. We say that S is regular if and only if:

(Def. 11) For every instruction type T of S holds $\mathrm{AddressParts}\,T$ is product-like.

Next we state the proposition

(19) For every instruction type T of $\mathrm{STC}(N)$ holds $\mathrm{AddressParts}\,T = \{0\}$.

Let N be a set with non empty elements. Observe that $\mathrm{STC}(N)$ is homogeneous explicit-jump-instruction and regular and has ins-loc-in-jump.

Let N be a set with non empty elements. Note that there exists a von Neumann definite AMI over N which is standard, halting, realistic, steady-programmed, programmable, explicit-jump-instruction, homogeneous, and regular and has non trivial instruction locations and ins-loc-in-jump.

Let N be a set with non empty elements, let S be a regular AMI over N, and let T be an instruction type of S. Observe that AddressParts T is product-like.

Let N be a set with non empty elements, let S be a homogeneous AMI over N, and let T be an instruction type of S. Observe that AddressParts T has common domain.

Next we state the proposition

(20) Let S be a homogeneous AMI over N, I be an instruction of S, and x be a set. Suppose $x \in \mathrm{dom}\,\mathrm{AddressPart}(I)$. Suppose $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(I)}(x) =$ the instruction locations of S. Then $(\mathrm{AddressPart}(I))(x)$ is an instruction-location of S.

Let N be a set with non empty elements and let S be an explicit-jump-instruction von Neumann definite AMI over N. Note that every instruction of S is explicit-jump-instruction.

Let N be a set with non empty elements and let S be a von Neumann definite AMI over N with ins-loc-in-jump. Observe that every instruction of S has ins-loc-in-jump.

The following proposition is true

(21) Let S be a realistic von Neumann definite AMI over N with non trivial instruction locations and I be an instruction of S. If I is halting, then $\mathrm{JUMP}(I)$ is empty.

Let N be a set with non empty elements, let S be a halting realistic von Neumann definite AMI over N with non trivial instruction locations, and let I be a halting instruction of S. One can verify that $\mathrm{JUMP}(I)$ is empty.

Let N be a set with non empty elements and let S be a von Neumann definite AMI over N with non trivial instruction locations. Observe that there exists a finite partial state of S which is non trivial and programmed.

Let N be a set with non empty elements and let S be a standard halting von Neumann definite AMI over N. One can verify that every non empty programmed finite partial state of S which is trivial is also unique-halt.

Let N be a set, let S be an AMI over N, and let I be an instruction of S. We say that I is instruction location free if and only if:

(Def. 12) For every set x such that $x \in \mathrm{dom}\,\mathrm{AddressPart}(I)$ holds $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(I)}(x) \neq$ the instruction locations of S.

The following propositions are true:

(22) Let S be a halting explicit-jump-instruction realistic von Neumann definite AMI over N with non trivial instruction locations and I be an instruction of S. If I is instruction location free, then $\mathrm{JUMP}(I)$ is empty.

(23) Let S be a realistic von Neumann definite AMI over N with ins-loc-in-jump and non trivial instruction locations and I be an instruction of S. If I is halting, then I is instruction location free.

Let N be a set with non empty elements and let S be a realistic von Neumann definite AMI over N with ins-loc-in-jump and non trivial instruction locations. Observe that every instruction of S which is halting is also instruction location free.

We now state the proposition

(24) Let S be a standard von Neumann definite AMI over N with ins-loc-in-jump and I be an instruction of S. If I is sequential, then I is instruction location free.

Let N be a set with non empty elements and let S be a standard von Neumann definite AMI over N with ins-loc-in-jump. One can check that every instruction of S which is sequential is also instruction location free.

Let N be a set with non empty elements and let S be a standard halting von Neumann definite AMI over N. The functor $\mathrm{Stop}\, S$ yielding a finite partial state of S is defined by:

(Def. 13) $\mathrm{Stop}\, S = \mathrm{il}_S(0) \overset{\cdot}{\longmapsto} \mathbf{halt}_S$.

Let N be a set with non empty elements and let S be a standard halting von Neumann definite AMI over N. Note that $\mathrm{Stop}\, S$ is lower non empty programmed and trivial.

Let N be a set with non empty elements and let S be a standard realistic halting von Neumann definite AMI over N. One can check that $\mathrm{Stop}\, S$ is closed.

Let N be a set with non empty elements and let S be a standard halting steady-programmed von Neumann definite AMI over N. Note that $\mathrm{Stop}\, S$ is autonomic.

We now state three propositions:

(25) For every standard halting von Neumann definite AMI S over N holds $\mathrm{card}\, \mathrm{Stop}\, S = 1$.

(26) Let S be a standard halting von Neumann definite AMI over N and F be a pre-Macro of S. If $\mathrm{card}\, F = 1$, then $F = \mathrm{Stop}\, S$.

(27) For every standard halting von Neumann definite AMI S over N holds $\mathrm{LastLoc}\, \mathrm{Stop}\, S = \mathrm{il}_S(0)$.

Let N be a set with non empty elements and let S be a standard halting von Neumann definite AMI over N. Note that $\mathrm{Stop}\, S$ is halt-ending and unique-halt.

Let N be a set with non empty elements and let S be a standard halting von Neumann definite AMI over N. Then $\mathrm{Stop}\, S$ is a pre-Macro of S.

4. On the Composition of Macro Instructions

Let N be a set with non empty elements, let S be a homogeneous regular standard von Neumann definite AMI over N, let I be an element of the instructions of S, and let k be a natural number. The functor $\mathrm{IncAddr}(I,k)$ yielding an instruction of S is defined by the conditions (Def. 14).

(Def. 14)(i) $\mathrm{InsCode}(\mathrm{IncAddr}(I,k)) = \mathrm{InsCode}(I)$,

(ii) $\mathrm{dom}\,\mathrm{AddressPart}(\mathrm{IncAddr}(I,k)) = \mathrm{dom}\,\mathrm{AddressPart}(I)$, and

(iii) for every set n such that $n \in \mathrm{dom}\,\mathrm{AddressPart}(I)$ holds if $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(I)}(n) =$ the instruction locations of S, then there exists an instruction-location f of S such that $f = (\mathrm{AddressPart}(I))(n)$ and $(\mathrm{AddressPart}(\mathrm{IncAddr}(I,k)))(n) = \mathrm{il}_S(k + \mathrm{locnum}(f))$ and if $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(I)}(n) \neq$ the instruction locations of S, then $(\mathrm{AddressPart}(\mathrm{IncAddr}(I,k)))(n) = (\mathrm{AddressPart}(I))(n)$.

Next we state three propositions:

(28) Let S be a homogeneous regular standard von Neumann definite AMI over N and I be an element of the instructions of S. Then $\mathrm{IncAddr}(I,0) = I$.

(29) Let S be a homogeneous regular standard von Neumann definite AMI over N and I be an instruction of S. If I is instruction location free, then $\mathrm{IncAddr}(I,k) = I$.

(30) Let S be a halting standard realistic homogeneous regular von Neumann definite AMI over N with ins-loc-in-jump. Then $\mathrm{IncAddr}(\mathbf{halt}_S,k) = \mathbf{halt}_S$.

Let N be a set with non empty elements, let S be a halting standard realistic homogeneous regular von Neumann definite AMI over N with ins-loc-in-jump, let I be a halting instruction of S, and let k be a natural number. Observe that $\mathrm{IncAddr}(I,k)$ is halting.

We now state several propositions:

(31) Let S be a homogeneous regular standard von Neumann definite AMI over N and I be an instruction of S. Then $\mathrm{AddressParts}\,\mathrm{InsCode}(I) = \mathrm{AddressParts}\,\mathrm{InsCode}(\mathrm{IncAddr}(I,k))$.

(32) Let S be a homogeneous regular standard von Neumann definite AMI over N and I, J be instructions of S. Given a natural number k such that $\mathrm{IncAddr}(I,k) = \mathrm{IncAddr}(J,k)$. Suppose $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(I)}(x) =$ the instruction locations of S. Then $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(J)}(x) =$ the instruction locations of S.

(33) Let S be a homogeneous regular standard von Neumann definite AMI over N and I, J be instructions of S. Given a natural number k such that $\mathrm{IncAddr}(I,k) = \mathrm{IncAddr}(J,k)$. Suppose $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(I)}(x) \neq$ the

instruction locations of S. Then $\prod_{\mathrm{AddressParts\,InsCode}(J)}(x) \neq$ the instruction locations of S.

(34) Let S be a homogeneous regular standard von Neumann definite AMI over N and I, J be instructions of S. If there exists a natural number k such that $\mathrm{IncAddr}(I,k) = \mathrm{IncAddr}(J,k)$, then $I = J$.

(35) Let S be a homogeneous regular standard halting realistic von Neumann definite AMI over N with ins-loc-in-jump and I be an instruction of S. If $\mathrm{IncAddr}(I,k) = \mathbf{halt}_S$, then $I = \mathbf{halt}_S$.

(36) Let S be a homogeneous regular standard halting realistic von Neumann definite AMI over N with ins-loc-in-jump and I be an instruction of S. If I is sequential, then $\mathrm{IncAddr}(I,k)$ is sequential.

(37) Let S be a homogeneous regular standard von Neumann definite AMI over N and I be an instruction of S. Then $\mathrm{IncAddr}(\mathrm{IncAddr}(I,k),m) = \mathrm{IncAddr}(I,k+m)$.

Let N be a set with non empty elements, let S be a homogeneous regular standard von Neumann definite AMI over N, let p be a programmed finite partial state of S, and let k be a natural number. The functor $\mathrm{IncAddr}(p,k)$ yields a finite partial state of S and is defined as follows:

(Def. 15) $\mathrm{dom}\,\mathrm{IncAddr}(p,k) = \mathrm{dom}\,p$ and for every natural number m such that $\mathrm{il}_S(m) \in \mathrm{dom}\,p$ holds $(\mathrm{IncAddr}(p,k))(\mathrm{il}_S(m)) = \mathrm{IncAddr}(\pi_{\mathrm{il}_S(m)}p, k)$.

Let N be a set with non empty elements, let S be a homogeneous regular standard von Neumann definite AMI over N, let F be a programmed finite partial state of S, and let k be a natural number. One can check that $\mathrm{IncAddr}(F,k)$ is programmed.

Let N be a set with non empty elements, let S be a homogeneous regular standard von Neumann definite AMI over N, let F be an empty programmed finite partial state of S, and let k be a natural number. One can verify that $\mathrm{IncAddr}(F,k)$ is empty.

Let N be a set with non empty elements, let S be a homogeneous regular standard von Neumann definite AMI over N, let F be a non empty programmed finite partial state of S, and let k be a natural number. One can verify that $\mathrm{IncAddr}(F,k)$ is non empty.

Let N be a set with non empty elements, let S be a homogeneous regular standard von Neumann definite AMI over N, let F be a lower programmed finite partial state of S, and let k be a natural number. One can verify that $\mathrm{IncAddr}(F,k)$ is lower.

The following propositions are true:

(38) Let S be a homogeneous regular standard von Neumann definite AMI over N and F be a programmed finite partial state of S. Then $\mathrm{IncAddr}(F,0) = F$.

(39) Let S be a homogeneous regular standard von Neumann definite AMI over N and F be a lower programmed finite partial state of S. Then $\mathrm{IncAddr}(\mathrm{IncAddr}(F, k), m) = \mathrm{IncAddr}(F, k + m)$.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, let p be a finite partial state of S, and let k be a natural number. The functor $\mathrm{Shift}(p, k)$ yielding a finite partial state of S is defined by the conditions (Def. 16).

(Def. 16)(i) $\operatorname{dom}\mathrm{Shift}(p, k) = \{\mathrm{il}_S(m + k); m$ ranges over natural numbers: $\mathrm{il}_S(m) \in \operatorname{dom} p\}$, and

(ii) for every natural number m such that $\mathrm{il}_S(m) \in \operatorname{dom} p$ holds $(\mathrm{Shift}(p, k))(\mathrm{il}_S(m + k)) = p(\mathrm{il}_S(m))$.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, let F be a finite partial state of S, and let k be a natural number. Note that $\mathrm{Shift}(F, k)$ is programmed.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, let F be an empty finite partial state of S, and let k be a natural number. One can check that $\mathrm{Shift}(F, k)$ is empty.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, let F be a non empty programmed finite partial state of S, and let k be a natural number. One can check that $\mathrm{Shift}(F, k)$ is non empty.

We now state four propositions:

(40) Let S be a standard von Neumann definite AMI over N and F be a programmed finite partial state of S. Then $\mathrm{Shift}(F, 0) = F$.

(41) Let S be a standard von Neumann definite AMI over N, F be a finite partial state of S, and k be a natural number. If $k > 0$, then $\mathrm{il}_S(0) \notin \operatorname{dom}\mathrm{Shift}(F, k)$.

(42) Let S be a standard von Neumann definite AMI over N and F be a finite partial state of S. Then $\mathrm{Shift}(\mathrm{Shift}(F, m), k) = \mathrm{Shift}(F, m + k)$.

(43) Let S be a standard von Neumann definite AMI over N and F be a programmed finite partial state of S. Then $\operatorname{dom} F \approx \operatorname{dom}\mathrm{Shift}(F, k)$.

Let N be a set with non empty elements, let S be a homogeneous regular standard von Neumann definite AMI over N, and let I be an instruction of S. We say that I is IC-good if and only if:

(Def. 17) For every natural number k and for all states s_1, s_2 of S such that $s_2 = s_1 {+\cdot} (\mathbf{IC}_S \overset{\cdot}{\longmapsto} (\mathbf{IC}_{(s_1)} + k))$ holds $\mathbf{IC}_{\mathrm{Exec}(I, s_1)} + k = \mathbf{IC}_{\mathrm{Exec}(\mathrm{IncAddr}(I,k), s_2)}$.

Let N be a set with non empty elements and let S be a homogeneous regular standard von Neumann definite AMI over N. We say that S is IC-good if and only if:

(Def. 18) Every instruction of S is IC-good.

Let N be a set with non empty elements, let S be an AMI over N, and let I be an instruction of S. We say that I is Exec-preserving if and only if the condition (Def. 19) is satisfied.

(Def. 19) Let s_1, s_2 be states of S. Suppose s_1 and s_2 are equal outside the instruction locations of S. Then $\mathrm{Exec}(I, s_1)$ and $\mathrm{Exec}(I, s_2)$ are equal outside the instruction locations of S.

Let N be a set with non empty elements and let S be an AMI over N. We say that S is Exec-preserving if and only if:

(Def. 20) Every instruction of S is Exec-preserving.

One can prove the following proposition

(44) Let S be a homogeneous regular standard von Neumann definite AMI over N with ins-loc-in-jump and I be an instruction of S. If I is sequential, then I is IC-good.

Let N be a set with non empty elements and let S be a homogeneous regular standard von Neumann definite AMI over N with ins-loc-in-jump. Observe that every instruction of S which is sequential is also IC-good.

The following proposition is true

(45) Let S be a homogeneous regular standard realistic von Neumann definite AMI over N with ins-loc-in-jump and I be an instruction of S. If I is halting, then I is IC-good.

Let N be a set with non empty elements and let S be a homogeneous regular standard realistic von Neumann definite AMI over N with ins-loc-in-jump. Note that every instruction of S which is halting is also IC-good.

The following proposition is true

(46) For every AMI S over N and for every instruction I of S such that I is halting holds I is Exec-preserving.

Let N be a set with non empty elements and let S be an AMI over N. Observe that every instruction of S which is halting is also Exec-preserving.

Let N be a set with non empty elements. One can verify that $\mathrm{STC}(N)$ is IC-good and Exec-preserving.

Let N be a set with non empty elements. One can check that there exists a homogeneous regular standard von Neumann definite AMI over N which is halting, realistic, steady-programmed, programmable, explicit-jump-instruction, IC-good, and Exec-preserving and has ins-loc-in-jump and non trivial instruction locations.

Let N be a set with non empty elements and let S be an IC-good homogeneous regular standard von Neumann definite AMI over N. Note that every instruction of S is IC-good.

Let N be a set with non empty elements and let S be an Exec-preserving AMI over N. Note that every instruction of S is Exec-preserving.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, and let F be a non empty programmed finite partial state of S. The functor $\mathrm{CutLastLoc}\,F$ yielding a finite partial state of S is defined by:

(Def. 21) $\mathrm{CutLastLoc}\,F = F \setminus (\mathrm{LastLoc}\,F \longmapsto F(\mathrm{LastLoc}\,F))$.

The following propositions are true:

(47) Let S be a standard von Neumann definite AMI over N and F be a non empty programmed finite partial state of S. Then $\mathrm{dom}\,\mathrm{CutLastLoc}\,F = \mathrm{dom}\,F \setminus \{\mathrm{LastLoc}\,F\}$.

(48) Let S be a standard von Neumann definite AMI over N and F be a non empty programmed finite partial state of S. Then $\mathrm{dom}\,F = \mathrm{dom}\,\mathrm{CutLastLoc}\,F \cup \{\mathrm{LastLoc}\,F\}$.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, and let F be a non empty trivial programmed finite partial state of S. Note that $\mathrm{CutLastLoc}\,F$ is empty.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, and let F be a non empty programmed finite partial state of S. Observe that $\mathrm{CutLastLoc}\,F$ is programmed.

Let N be a set with non empty elements, let S be a standard von Neumann definite AMI over N, and let F be a lower non empty programmed finite partial state of S. Note that $\mathrm{CutLastLoc}\,F$ is lower.

We now state three propositions:

(49) Let S be a standard von Neumann definite AMI over N and F be a non empty programmed finite partial state of S. Then $\mathrm{card}\,\mathrm{CutLastLoc}\,F = \mathrm{card}\,F - 1$.

(50) Let S be a homogeneous regular standard von Neumann definite AMI over N, F be a lower non empty programmed finite partial state of S, and G be a non empty programmed finite partial state of S. Then $\mathrm{dom}\,\mathrm{CutLastLoc}\,F \cap \mathrm{dom}\,\mathrm{Shift}(\mathrm{IncAddr}(G, \mathrm{card}\,F -' 1), \mathrm{card}\,F -' 1) = \emptyset$.

(51) Let S be a standard halting von Neumann definite AMI over N, F be a unique-halt lower non empty programmed finite partial state of S, and I be an instruction-location of S. If $I \in \mathrm{dom}\,\mathrm{CutLastLoc}\,F$, then $(\mathrm{CutLastLoc}\,F)(I) \neq \mathbf{halt}_S$.

Let N be a set with non empty elements, let S be a homogeneous regular standard von Neumann definite AMI over N, and let F, G be non empty programmed finite partial states of S. The functor $F;\, G$ yields a finite partial state of S and is defined by:

(Def. 22) $F;\, G = \mathrm{CutLastLoc}\,F {+\cdot}\, \mathrm{Shift}(\mathrm{IncAddr}(G, \mathrm{card}\,F -' 1), \mathrm{card}\,F -' 1)$.

Let N be a set with non empty elements, let S be a homogeneous regular standard von Neumann definite AMI over N, and let F, G be non empty programmed finite partial states of S. Note that F; G is non empty and programmed.

We now state the proposition

(52) Let S be a homogeneous regular standard von Neumann definite AMI over N and F, G be lower non empty programmed finite partial states of S. Then $\mathrm{card}(F;\, G) = (\mathrm{card}\, F + \mathrm{card}\, G) - 1$ and $\mathrm{card}(F;\, G) = (\mathrm{card}\, F + \mathrm{card}\, G) -' 1$.

Let N be a set with non empty elements, let S be a homogeneous regular standard von Neumann definite AMI over N, and let F, G be lower non empty programmed finite partial states of S. Observe that F; G is lower.

We now state four propositions:

(53) Let S be a homogeneous regular standard von Neumann definite AMI over N and F, G be lower non empty programmed finite partial states of S. Then $\mathrm{dom}\, F \subseteq \mathrm{dom}(F;\, G)$.

(54) Let S be a homogeneous regular standard von Neumann definite AMI over N and F, G be lower non empty programmed finite partial states of S. Then $\mathrm{CutLastLoc}\, F \subseteq \mathrm{CutLastLoc}\, F;\, G$.

(55) Let S be a homogeneous regular standard von Neumann definite AMI over N and F, G be lower non empty programmed finite partial states of S. Then $(F;\, G)(\mathrm{LastLoc}\, F) = (\mathrm{IncAddr}(G, \mathrm{card}\, F -' 1))(\mathrm{il}_S(0))$.

(56) Let S be a homogeneous regular standard von Neumann definite AMI over N, F, G be lower non empty programmed finite partial states of S, and f be an instruction-location of S. If $\mathrm{locnum}(f) < \mathrm{card}\, F - 1$, then $(\mathrm{IncAddr}(F, \mathrm{card}\, F -' 1))(f) = (\mathrm{IncAddr}(F;\, G, \mathrm{card}\, F -' 1))(f)$.

Let N be a set with non empty elements, let S be a homogeneous regular standard realistic halting steady-programmed von Neumann definite AMI over N with ins-loc-in-jump, and let F, G be halt-ending lower non empty programmed finite partial states of S. Observe that F; G is halt-ending.

Let N be a set with non empty elements, let S be a homogeneous regular standard realistic halting steady-programmed von Neumann definite AMI over N with ins-loc-in-jump, and let F, G be halt-ending unique-halt lower non empty programmed finite partial states of S. Observe that F; G is unique-halt.

Let N be a set with non empty elements, let S be a homogeneous regular standard realistic halting steady-programmed von Neumann definite AMI over N with ins-loc-in-jump, and let F, G be pre-Macros of S. Then F; G is a pre-Macro of S.

Let N be a set with non empty elements, let S be a realistic halting steady-programmed IC-good Exec-preserving homogeneous regular standard von Neumann definite AMI over N, and let F, G be closed lower non empty programmed finite partial states of S. Observe that F; G is closed.

We now state several propositions:

(57) Let S be a homogeneous regular standard halting realistic von Neumann definite AMI over N with ins-loc-in-jump. Then $\mathrm{IncAddr}(\mathrm{Stop}\,S, k) = \mathrm{Stop}\,S$.

(58) For every standard halting von Neumann definite AMI S over N holds $\mathrm{Shift}(\mathrm{Stop}\,S, k) = \mathrm{il}_S(k) \overset{.}{\longmapsto} \mathbf{halt}_S$.

(59) Let S be a homogeneous regular standard halting realistic von Neumann definite AMI over N with ins-loc-in-jump and F be a pre-Macro of S. Then $F;\ \mathrm{Stop}\,S = F$.

(60) Let S be a homogeneous regular standard halting von Neumann definite AMI over N and F be a pre-Macro of S. Then $\mathrm{Stop}\,S;\ F = F$.

(61) Let S be a homogeneous regular standard realistic halting steady-programmed von Neumann definite AMI over N with ins-loc-in-jump and F, G, H be pre-Macros of S. Then $(F;\ G);\ H = F;\ (G;\ H)$.

References

[1] Grzegorz Bancerek. Cardinal numbers. *Formalized Mathematics*, 1(**2**):377–382, 1990.

[2] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[3] Grzegorz Bancerek. König's theorem. *Formalized Mathematics*, 1(**3**):589–593, 1990.

[4] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[5] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[6] Czesław Byliński. The modification of a function by a function and the iteration of the composition of a function. *Formalized Mathematics*, 1(**3**):521–527, 1990.

[7] Czesław Byliński. Some basic properties of sets. *Formalized Mathematics*, 1(**1**):47–53, 1990.

[8] Agata Darmochwał. Finite sets. *Formalized Mathematics*, 1(**1**):165–167, 1990.

[9] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[10] Beata Madras. Products of many sorted algebras. *Formalized Mathematics*, 5(**1**):55–60, 1996.

[11] Yatsuka Nakamura and Andrzej Trybulec. A mathematical model of CPU. *Formalized Mathematics*, 3(**2**):151–160, 1992.

[12] Takaya Nishiyama and Yasuho Mizuhara. Binary arithmetics. *Formalized Mathematics*, 4(**1**):83–86, 1993.

[13] Beata Padlewska. Families of sets. *Formalized Mathematics*, 1(**1**):147–152, 1990.

[14] Andrzej Trybulec. Domains and their Cartesian products. *Formalized Mathematics*, 1(**1**):115–122, 1990.

[15] Andrzej Trybulec. Function domains and Frænkel operator. *Formalized Mathematics*, 1(**3**):495–500, 1990.

[16] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[17] Andrzej Trybulec and Yatsuka Nakamura. Some remarks on the simple concrete model of computer. *Formalized Mathematics*, 4(**1**):51–56, 1993.

[18] Andrzej Trybulec, Piotr Rudnicki, and Artur Korniłowicz. Standard ordering of instruction locations. *Formalized Mathematics*, 9(**2**):291–301, 2001.

[19] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[20] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

[21] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

Received April 14, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

The Properties of Instructions of SCM over Ring

Artur Korniłowicz
University of Białystok

MML Identifier: SCMRING3.

The papers [16], [9], [11], [12], [15], [19], [2], [3], [5], [6], [4], [1], [20], [21], [17], [8], [7], [13], [18], [14], and [10] provide the terminology and notation for this paper.

For simplicity, we adopt the following convention: R denotes a good ring, r denotes an element of the carrier of R, a, b denote Data-Locations of R, i_1, i_2, i_3 denote instruction-locations of $\mathbf{SCM}(R)$, I denotes an instruction of $\mathbf{SCM}(R)$, s_1, s_2 denote states of $\mathbf{SCM}(R)$, T denotes an instruction type of $\mathbf{SCM}(R)$, and k denotes a natural number.

Let us note that $\mathbb{Z}$ is infinite.

One can verify that INT.Ring is infinite and good.

Let us mention that there exists a 1-sorted structure which is strict and infinite.

Let us mention that there exists a ring which is strict, infinite, and good.

We now state the proposition

(1) ObjectKind(a) = the carrier of R.

Let R be a good ring, let l_1, l_2 be Data-Locations of R, and let a, b be elements of R. Then $[l_1 \longmapsto a, l_2 \longmapsto b]$ is a finite partial state of $\mathbf{SCM}(R)$.

We now state a number of propositions:

(2) $a \notin$ the instruction locations of $\mathbf{SCM}(R)$.

(3) $a \neq \mathbf{IC}_{\mathbf{SCM}(R)}$.

(4) Data-Loc$_{\rm SCM}$ $\neq$ the instruction locations of $\mathbf{SCM}(R)$.

(5) For every object o of $\mathbf{SCM}(R)$ holds $o = \mathbf{IC}_{\mathbf{SCM}(R)}$ or $o \in$ the instruction locations of $\mathbf{SCM}(R)$ or o is a Data-Location of R.

(6) If $i_2 \neq i_3$, then Next$(i_2) \neq$ Next(i_3).

ISSN 1426–2630

(7) If s_1 and s_2 are equal outside the instruction locations of $\mathbf{SCM}(R)$, then $s_1(a) = s_2(a)$.

(8) $\mathrm{InsCode}(\mathbf{halt}_{\mathbf{SCM}(R)}) = 0$.

(9) $\mathrm{InsCode}(a{:=}b) = 1$.

(10) $\mathrm{InsCode}(\mathrm{AddTo}(a, b)) = 2$.

(11) $\mathrm{InsCode}(\mathrm{SubFrom}(a, b)) = 3$.

(12) $\mathrm{InsCode}(\mathrm{MultBy}(a, b)) = 4$.

(13) $\mathrm{InsCode}(a{:=}r) = 5$.

(14) $\mathrm{InsCode}(\mathrm{goto}\ i_2) = 6$.

(15) $\mathrm{InsCode}(\mathbf{if}\ a = 0\ \mathbf{goto}\ i_2) = 7$.

(16) If $\mathrm{InsCode}(I) = 0$, then $I = \mathbf{halt}_{\mathbf{SCM}(R)}$.

(17) If $\mathrm{InsCode}(I) = 1$, then there exist a, b such that $I = a{:=}b$.

(18) If $\mathrm{InsCode}(I) = 2$, then there exist a, b such that $I = \mathrm{AddTo}(a, b)$.

(19) If $\mathrm{InsCode}(I) = 3$, then there exist a, b such that $I = \mathrm{SubFrom}(a, b)$.

(20) If $\mathrm{InsCode}(I) = 4$, then there exist a, b such that $I = \mathrm{MultBy}(a, b)$.

(21) If $\mathrm{InsCode}(I) = 5$, then there exist a, r such that $I = a{:=}r$.

(22) If $\mathrm{InsCode}(I) = 6$, then there exists i_3 such that $I = \mathrm{goto}\ i_3$.

(23) If $\mathrm{InsCode}(I) = 7$, then there exist a, i_2 such that $I = \mathbf{if}\ a = 0\ \mathbf{goto}\ i_2$.

(24) $\mathrm{AddressPart}(\mathbf{halt}_{\mathbf{SCM}(R)}) = \varepsilon$.

(25) $\mathrm{AddressPart}(a{:=}b) = \langle a, b \rangle$.

(26) $\mathrm{AddressPart}(\mathrm{AddTo}(a, b)) = \langle a, b \rangle$.

(27) $\mathrm{AddressPart}(\mathrm{SubFrom}(a, b)) = \langle a, b \rangle$.

(28) $\mathrm{AddressPart}(\mathrm{MultBy}(a, b)) = \langle a, b \rangle$.

(29) $\mathrm{AddressPart}(a{:=}r) = \langle a, r \rangle$.

(30) $\mathrm{AddressPart}(\mathrm{goto}\ i_2) = \langle i_2 \rangle$.

(31) $\mathrm{AddressPart}(\mathbf{if}\ a = 0\ \mathbf{goto}\ i_2) = \langle i_2, a \rangle$.

(32) If $T = 0$, then $\mathrm{AddressParts}\, T = \{0\}$.

Let us consider R, T. Observe that $\mathrm{AddressParts}\, T$ is non empty.

We now state a number of propositions:

(33) If $T = 1$, then $\mathrm{dom} \prod_{\mathrm{AddressParts}\, T} = \{1, 2\}$.

(34) If $T = 2$, then $\mathrm{dom} \prod_{\mathrm{AddressParts}\, T} = \{1, 2\}$.

(35) If $T = 3$, then $\mathrm{dom} \prod_{\mathrm{AddressParts}\, T} = \{1, 2\}$.

(36) If $T = 4$, then $\mathrm{dom} \prod_{\mathrm{AddressParts}\, T} = \{1, 2\}$.

(37) If $T = 5$, then $\mathrm{dom} \prod_{\mathrm{AddressParts}\, T} = \{1, 2\}$.

(38) If $T = 6$, then $\mathrm{dom} \prod_{\mathrm{AddressParts}\, T} = \{1\}$.

(39) If $T = 7$, then $\mathrm{dom} \prod_{\mathrm{AddressParts}\, T} = \{1, 2\}$.

(40) $\prod_{\mathrm{AddressParts}\, \mathrm{InsCode}(a{:=}b)}(1) = \text{Data-Loc}_{\mathrm{SCM}}$.

(41) $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(a{:=}b)}(2) = \text{Data-Loc}_{\mathrm{SCM}}$.

(42) $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(\mathrm{AddTo}(a,b))}(1) = \text{Data-Loc}_{\mathrm{SCM}}$.

(43) $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(\mathrm{AddTo}(a,b))}(2) = \text{Data-Loc}_{\mathrm{SCM}}$.

(44) $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(\mathrm{SubFrom}(a,b))}(1) = \text{Data-Loc}_{\mathrm{SCM}}$.

(45) $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(\mathrm{SubFrom}(a,b))}(2) = \text{Data-Loc}_{\mathrm{SCM}}$.

(46) $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(\mathrm{MultBy}(a,b))}(1) = \text{Data-Loc}_{\mathrm{SCM}}$.

(47) $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(\mathrm{MultBy}(a,b))}(2) = \text{Data-Loc}_{\mathrm{SCM}}$.

(48) $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(a{:=}r)}(1) = \text{Data-Loc}_{\mathrm{SCM}}$.

(49) $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(a{:=}r)}(2) =$ the carrier of R.

(50) $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(\mathrm{goto}\ i_2)}(1) =$ the instruction locations of $\mathbf{SCM}(R)$.

(51) $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(\mathbf{if}\ a=0\ \mathbf{goto}\ i_2)}(1) =$ the instruction locations of $\mathbf{SCM}(R)$.

(52) $\prod_{\mathrm{AddressParts}\,\mathrm{InsCode}(\mathbf{if}\ a=0\ \mathbf{goto}\ i_2)}(2) = \text{Data-Loc}_{\mathrm{SCM}}$.

(53) $\mathrm{NIC}(\mathbf{halt}_{\mathbf{SCM}(R)}, i_1) = \{i_1\}$.

Let us consider R. One can check that $\mathrm{JUMP}(\mathbf{halt}_{\mathbf{SCM}(R)})$ is empty.

Next we state the proposition

(54) $\mathrm{NIC}(a{:=}b, i_1) = \{\mathrm{Next}(i_1)\}$.

Let us consider R, a, b. Observe that $\mathrm{JUMP}(a{:=}b)$ is empty.

We now state the proposition

(55) $\mathrm{NIC}(\mathrm{AddTo}(a,b), i_1) = \{\mathrm{Next}(i_1)\}$.

Let us consider R, a, b. One can check that $\mathrm{JUMP}(\mathrm{AddTo}(a,b))$ is empty.

One can prove the following proposition

(56) $\mathrm{NIC}(\mathrm{SubFrom}(a,b), i_1) = \{\mathrm{Next}(i_1)\}$.

Let us consider R, a, b. Note that $\mathrm{JUMP}(\mathrm{SubFrom}(a,b))$ is empty.

Next we state the proposition

(57) $\mathrm{NIC}(\mathrm{MultBy}(a,b), i_1) = \{\mathrm{Next}(i_1)\}$.

Let us consider R, a, b. One can verify that $\mathrm{JUMP}(\mathrm{MultBy}(a,b))$ is empty.

One can prove the following proposition

(58) $\mathrm{NIC}(a{:=}r, i_1) = \{\mathrm{Next}(i_1)\}$.

Let us consider R, a, r. Note that $\mathrm{JUMP}(a{:=}r)$ is empty.

The following propositions are true:

(59) $\mathrm{NIC}(\mathrm{goto}\ i_2, i_1) = \{i_2\}$.

(60) $\mathrm{JUMP}(\mathrm{goto}\ i_2) = \{i_2\}$.

Let us consider R, i_2. Note that $\mathrm{JUMP}(\mathrm{goto}\ i_2)$ is non empty and trivial.

We now state two propositions:

(61) $i_2 \in \mathrm{NIC}(\mathbf{if}\ a = 0\ \mathbf{goto}\ i_2, i_1)$ and $\mathrm{NIC}(\mathbf{if}\ a = 0\ \mathbf{goto}\ i_2, i_1) \subseteq \{i_2, \mathrm{Next}(i_1)\}$.

(62) $\mathrm{JUMP}(\mathbf{if}\ a = 0\ \mathbf{goto}\ i_2) = \{i_2\}$.

Let us consider R, a, i_2. Observe that JUMP(**if** $a = 0$ **goto** i_2) is non empty and trivial.

One can prove the following two propositions:

(63) $\mathrm{SUCC}(i_1) = \{i_1, \mathrm{Next}(i_1)\}$.

(64) Let f be a function from $\mathbb{N}$ into the instruction locations of $\mathbf{SCM}(R)$. Suppose that for every natural number k holds $f(k) = \mathbf{i}_k$. Then

(i) f is bijective, and

(ii) for every natural number k holds $f(k+1) \in \mathrm{SUCC}(f(k))$ and for every natural number j such that $f(j) \in \mathrm{SUCC}(f(k))$ holds $k \leqslant j$.

Let us consider R. Note that $\mathbf{SCM}(R)$ is standard.

Next we state three propositions:

(65) $\mathrm{il}_{\mathbf{SCM}(R)}(k) = \mathbf{i}_k$.

(66) $\mathrm{Next}(\mathrm{il}_{\mathbf{SCM}(R)}(k)) = \mathrm{il}_{\mathbf{SCM}(R)}(k+1)$.

(67) $\mathrm{Next}(i_1) = \mathrm{NextLoc}\, i_1$.

Let R be a good ring and let k be a natural number. The functor $\mathrm{dl}_R(k)$ yields a Data-Location of R and is defined as follows:

(Def. 1) $\mathrm{dl}_R(k) = \mathbf{d}_k$.

Let us consider R. Observe that InsCode($\mathbf{halt}_{\mathbf{SCM}(R)}$) is jump-only.

Let us consider R. Note that $\mathbf{halt}_{\mathbf{SCM}(R)}$ is jump-only.

Let us consider R, i_2. Note that InsCode(goto i_2) is jump-only.

Let us consider R, i_2. One can check that goto i_2 is jump-only.

Let us consider R, a, i_2. Observe that InsCode(**if** $a = 0$ **goto** i_2) is jump-only.

Let us consider R, a, i_2. Note that **if** $a = 0$ **goto** i_2 is jump-only.

In the sequel S denotes a non trivial good ring, p, q denote Data-Locations of S, and w denotes an element of the carrier of S.

Let us consider S, p, q. One can check that InsCode(p:=q) is non jump-only.

Let us consider S, p, q. One can check that p:=q is non jump-only.

Let us consider S, p, q. Observe that InsCode(AddTo(p, q)) is non jump-only.

Let us consider S, p, q. Note that AddTo(p, q) is non jump-only.

Let us consider S, p, q. Note that InsCode(SubFrom(p, q)) is non jump-only.

Let us consider S, p, q. Note that SubFrom(p, q) is non jump-only.

Let us consider S, p, q. Observe that InsCode(MultBy(p, q)) is non jump-only.

Let us consider S, p, q. One can verify that MultBy(p, q) is non jump-only.

Let us consider S, p, w. Note that InsCode(p:=w) is non jump-only.

Let us consider S, p, w. Note that p:=w is non jump-only.

Let us consider R, a, b. Observe that a:=b is sequential.

Let us consider R, a, b. Observe that AddTo(a, b) is sequential.

Let us consider R, a, b. Note that SubFrom(a, b) is sequential.

Let us consider R, a, b. One can verify that MultBy(a, b) is sequential.

Let us consider R, a, r. Note that $a{:=}r$ is sequential.

Let us consider R, i_2. One can check that goto i_2 is non sequential.

Let us consider R, a, i_2. Observe that **if** $a = 0$ **goto** i_2 is non sequential.

Let us consider R, i_2. Note that goto i_2 is non instruction location free.

Let us consider R, a, i_2. Note that **if** $a = 0$ **goto** i_2 is non instruction location free.

Let us consider R. One can check that **SCM**(R) is homogeneous and explicit-jump-instruction and has ins-loc-in-jump.

Let us consider R. Observe that **SCM**(R) is regular.

Next we state two propositions:

(68) IncAddr(goto i_2, k) = goto $\mathrm{il}_{\mathbf{SCM}(R)}(\mathrm{locnum}(i_2) + k)$.

(69) IncAddr(**if** $a = 0$ **goto** i_2, k) = **if** $a = 0$ **goto** $\mathrm{il}_{\mathbf{SCM}(R)}(\mathrm{locnum}(i_2){+}k)$.

Let us consider R. One can check that **SCM**(R) is IC-good and Exec-preserving.

References

[1] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[2] Grzegorz Bancerek. König's theorem. *Formalized Mathematics*, 1(**3**):589–593, 1990.

[3] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[4] Grzegorz Bancerek and Andrzej Trybulec. Miscellaneous facts about functions. *Formalized Mathematics*, 5(**4**):485–492, 1996.

[5] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[6] Czesław Byliński. Functions from a set to a set. *Formalized Mathematics*, 1(**1**):153–164, 1990.

[7] Artur Korniłowicz. The basic properties of **SCM** over ring. *Formalized Mathematics*, 7(**2**):301–305, 1998.

[8] Artur Korniłowicz. The construction of **SCM** over ring. *Formalized Mathematics*, 7(**2**):295–300, 1998.

[9] Artur Korniłowicz. On the composition of macro instructions of standard computers. *Formalized Mathematics*, 9(**2**):303–316, 2001.

[10] Eugeniusz Kusak, Wojciech Leończuk, and Michał Muzalewski. Abelian groups, fields and vector spaces. *Formalized Mathematics*, 1(**2**):335–342, 1990.

[11] Yatsuka Nakamura and Andrzej Trybulec. A mathematical model of CPU. *Formalized Mathematics*, 3(**2**):151–160, 1992.

[12] Yatsuka Nakamura and Andrzej Trybulec. On a mathematical model of programs. *Formalized Mathematics*, 3(**2**):241–250, 1992.

[13] Yozo Toda. The formalization of simple graphs. *Formalized Mathematics*, 5(**1**):137–144, 1996.

[14] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[15] Andrzej Trybulec and Yatsuka Nakamura. Some remarks on the simple concrete model of computer. *Formalized Mathematics*, 4(**1**):51–56, 1993.

[16] Andrzej Trybulec, Piotr Rudnicki, and Artur Korniłowicz. Standard ordering of instruction locations. *Formalized Mathematics*, 9(**2**):291–301, 2001.

[17] Wojciech A. Trybulec. Vectors in real linear space. *Formalized Mathematics*, 1(**2**):291–296, 1990.

[18] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[19] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

[20] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

[21] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

Received April 14, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Basic Facts about Inaccessible and Measurable Cardinals

Josef Urban
Charles University
Praha

Summary. Inaccessible, strongly inaccessible and measurable cardinals are defined, and it is proved that a measurable cardinal is strongly inaccessible. Filters on sets are defined, some facts related to the section about cardinals are proved. Existence of the Ulam matrix on non-limit cardinals is proved.

MML Identifier: `CARD_FIL`.

The notation and terminology used here are introduced in the following papers: [13], [2], [1], [5], [9], [6], [7], [3], [4], [14], [10], [12], [11], and [8].

1. Some Facts about Filters and Ideals on Sets

One can verify that there exists an ordinal number which is limit.

Let X, Y be sets. Then $X \setminus Y$ is a subset of X.

We now state the proposition

(1) For every set x and for every infinite set X holds $\overline{\overline{\{x\}}} < \overline{\overline{X}}$.

Let X be an infinite set. Observe that $\overline{\overline{X}}$ is infinite.

The scheme *ElemProp* deals with a non empty set $\mathcal{A}$, a set $\mathcal{B}$, and a unary predicate $\mathcal{P}$, and states that:

$\mathcal{P}[\mathcal{B}]$

provided the following condition is met:

- $\mathcal{B} \in \{y; y$ ranges over elements of $\mathcal{A} : \mathcal{P}[y]\}$.

For simplicity, we follow the rules: N is a cardinal number, M is an aleph, X is a non empty set, Y, Z, Z_1, Z_2, Y_1, Y_2 are subsets of X, and S is a subset of 2^X.

One can prove the following proposition

(2)(i) $\{X\}$ is a non empty subset of 2^X,

(ii) $\emptyset \notin \{X\}$, and

(iii) for all Y_1, Y_2 holds if $Y_1 \in \{X\}$ and $Y_2 \in \{X\}$, then $Y_1 \cap Y_2 \in \{X\}$ and if $Y_1 \in \{X\}$ and $Y_1 \subseteq Y_2$, then $Y_2 \in \{X\}$.

Let us consider X. A non empty subset of 2^X is said to be a filter of X if:

(Def. 1) $\emptyset \notin$ it and for all Y_1, Y_2 holds if $Y_1 \in$ it and $Y_2 \in$ it, then $Y_1 \cap Y_2 \in$ it and if $Y_1 \in$ it and $Y_1 \subseteq Y_2$, then $Y_2 \in$ it.

The following propositions are true:

(3) Let F be a set. Then F is a filter of X if and only if the following conditions are satisfied:

(i) F is a non empty subset of 2^X,

(ii) $\emptyset \notin F$, and

(iii) for all Y_1, Y_2 holds if $Y_1 \in F$ and $Y_2 \in F$, then $Y_1 \cap Y_2 \in F$ and if $Y_1 \in F$ and $Y_1 \subseteq Y_2$, then $Y_2 \in F$.

(4) $\{X\}$ is a filter of X.

In the sequel F, F_1, F_2, U_1 denote filters of X.

The following propositions are true:

(5) $X \in F$.

(6) If $Y \in F$, then $X \setminus Y \notin F$.

(7) Let I be a non empty subset of 2^X. Suppose that for every Y holds $Y \in I$ iff $Y^{\mathrm{c}} \in F$. Then $X \notin I$ and for all Y_1, Y_2 holds if $Y_1 \in I$ and $Y_2 \in I$, then $Y_1 \cup Y_2 \in I$ and if $Y_1 \in I$ and $Y_2 \subseteq Y_1$, then $Y_2 \in I$.

Let us consider X, S. We introduce $\mathrm{dual}\, S$ as a synonym of S^{c}.

In the sequel S is a non empty subset of 2^X.

Let us consider X, S. One can verify that S^{c} is non empty.

One can prove the following two propositions:

(8) $\mathrm{dual}\, S = \{Y : Y^{\mathrm{c}} \in S\}$.

(9) $\mathrm{dual}\, S = \{Y^{\mathrm{c}} : Y \in S\}$.

Let us consider X. A non empty subset of 2^X is said to be an ideal of X if:

(Def. 2) $X \notin$ it and for all Y_1, Y_2 holds if $Y_1 \in$ it and $Y_2 \in$ it, then $Y_1 \cup Y_2 \in$ it and if $Y_1 \in$ it and $Y_2 \subseteq Y_1$, then $Y_2 \in$ it.

Let us consider X, F. Then $\mathrm{dual}\, F$ is an ideal of X.

In the sequel I is an ideal of X.

Next we state two propositions:

(10) For every Y holds $Y \notin F$ or $Y \notin \mathrm{dual}\, F$ and for every Y holds $Y \notin I$ or $Y \notin \mathrm{dual}\, I$.

(11) $\emptyset \in I$.

Let us consider X, N, S. We say that S is multiplicative with N if and only if:

(Def. 3) For every non empty set S_1 such that $S_1 \subseteq S$ and $\overline{\overline{S_1}} < N$ holds $\bigcap S_1 \in S$.

Let us consider X, N, S. We say that S is additive with N if and only if:

(Def. 4) For every non empty set S_1 such that $S_1 \subseteq S$ and $\overline{\overline{S_1}} < N$ holds $\bigcup S_1 \in S$.

Let us consider X, N, F. We introduce F is complete with N as a synonym of F is multiplicative with N.

Let us consider X, N, I. We introduce I is complete with N as a synonym of I is additive with N.

One can prove the following proposition

(12) If S is multiplicative with N, then dual S is additive with N.

Let us consider X, F. We say that F is uniform if and only if:

(Def. 5) For every Y such that $Y \in F$ holds $\overline{\overline{Y}} = \overline{\overline{X}}$.

We say that F is principal if and only if:

(Def. 6) There exists Y such that $Y \in F$ and for every Z such that $Z \in F$ holds $Y \subseteq Z$.

We say that F is an ultrafilter if and only if:

(Def. 7) For every Y holds $Y \in F$ or $X \setminus Y \in F$.

Let us consider X, F, Z. The functor Extend_Filter(F, Z) yields a non empty subset of 2^X and is defined as follows:

(Def. 8) $\mathrm{Extend\_Filter}(F, Z) = \{Y : \bigvee_{Y_2} (Y_2 \in \{Y_1 \cap Z : Y_1 \in F\} \ \wedge \ Y_2 \subseteq Y)\}$.

We now state two propositions:

(13) For every Z_1 holds $Z_1 \in \mathrm{Extend\_Filter}(F, Z)$ iff there exists Z_2 such that $Z_2 \in F$ and $Z_2 \cap Z \subseteq Z_1$.

(14) If for every Y_1 such that $Y_1 \in F$ holds $Y_1 \cap Z \neq \emptyset$, then $Z \in \mathrm{Extend\_Filter}(F, Z)$ and Extend_Filter(F, Z) is a filter of X and $F \subseteq \mathrm{Extend\_Filter}(F, Z)$.

In the sequel S denotes a subset of 2^X.

Let us consider X. The functor Filters X yielding a non empty subset of 2^{2^X} is defined by:

(Def. 9) $\mathrm{Filters}\, X = \{S : S$ is a filter of $X\}$.

We now state the proposition

(15) For every set S holds $S \in \mathrm{Filters}\, X$ iff S is a filter of X.

In the sequel F_3 is a non empty subset of Filters X.

One can prove the following propositions:

(16) If for all F_1, F_2 such that $F_1 \in F_3$ and $F_2 \in F_3$ holds $F_1 \subseteq F_2$ or $F_2 \subseteq F_1$, then $\bigcup F_3$ is a filter of X.

(17) For every F there exists U_1 such that $F \subseteq U_1$ and U_1 is an ultrafilter.

In the sequel X denotes an infinite set, Y denotes a subset of X, and F, U_1 denote filters of X.

Let us consider X. The functor Frechet_Filter X yielding a filter of X is defined by:

(Def. 10) Frechet_Filter $X = \{Y : \overline{\overline{X \setminus Y}} < \overline{\overline{X}}\}$.

Let us consider X. The functor Frechet_Ideal X yields an ideal of X and is defined as follows:

(Def. 11) Frechet_Ideal X = dual Frechet_Filter X.

One can prove the following propositions:

(18) $Y \in$ Frechet_Filter X iff $\overline{\overline{X \setminus Y}} < \overline{\overline{X}}$.

(19) $Y \in$ Frechet_Ideal X iff $\overline{\overline{Y}} < \overline{\overline{X}}$.

(20) If Frechet_Filter $X \subseteq F$, then F is uniform.

(21) If U_1 is uniform and an ultrafilter, then Frechet_Filter $X \subseteq U_1$.

Let us consider X. One can check that there exists a filter of X which is non principal and an ultrafilter.

Let us consider X. One can check that every filter of X which is uniform and an ultrafilter is also non principal.

Next we state two propositions:

(22) For every an ultrafilter filter F of X and for every Y holds $Y \in F$ iff $Y \notin$ dual F.

(23) If F is non principal and an ultrafilter and F is complete with $\overline{\overline{X}}$, then F is uniform.

2. Inaccessible and Measurable Cardinals, Ulam Matrix

We now state the proposition

(24) $N^+ \leqslant \overline{\mathbf{2}}^N$.

We say that Generalized Continuum Hypothesis holds if and only if:

(Def. 12) For every N holds $N^+ = \overline{\mathbf{2}}^N$.

Let I_1 be an aleph. We say that I_1 is inaccessible if and only if:

(Def. 13) I_1 is regular and limit.

We introduce I_1 is inaccessible cardinal as a synonym of I_1 is inaccessible.

Let us note that every aleph which is inaccessible is also regular and limit.

We now state the proposition

(25) $\aleph_0$ is inaccessible.

Let I_1 be an aleph. We say that I_1 is strong limit if and only if:

(Def. 14) For every N such that $N < I_1$ holds $\overline{\mathbf{2}}^N < I_1$.

We introduce I_1 is strong limit cardinal as a synonym of I_1 is strong limit.

Next we state two propositions:

(26) $\aleph_0$ is strong limit.

(27) If M is strong limit, then M is limit.

One can check that every aleph which is strong limit is also limit.

Next we state the proposition

(28) If Generalized Continuum Hypothesis holds, then if M is limit, then M is strong limit.

Let I_1 be an aleph. We say that I_1 is strongly inaccessible if and only if:

(Def. 15) I_1 is regular and strong limit.

We introduce I_1 is strongly inaccessible cardinal as a synonym of I_1 is strongly inaccessible.

Let us observe that every aleph which is strongly inaccessible is also regular and strong limit.

The following propositions are true:

(29) $\aleph_0$ is strongly inaccessible.

(30) If M is strongly inaccessible, then M is inaccessible.

Let us note that every aleph which is strongly inaccessible is also inaccessible.

Next we state the proposition

(31) If Generalized Continuum Hypothesis holds, then if M is inaccessible, then M is strongly inaccessible.

Let us consider M. We say that M is measurable if and only if:

(Def. 16) There exists a filter U_1 of M such that U_1 is complete with M and U_1 is non principal and an ultrafilter.

We introduce M is measurable cardinal as a synonym of M is measurable.

We now state two propositions:

(32) For every limit ordinal number A and for every set X such that $X \subseteq A$ holds if $\sup X = A$, then $\bigcup X = A$.

(33) If M is measurable, then M is regular.

Let us consider M. Note that M^+ is non limit.

Let us note that there exists a cardinal number which is non limit and infinite.

Let us observe that every aleph which is non limit is also regular.

Let M be a non limit cardinal number. The functor predecessor M yields a cardinal number and is defined as follows:

(Def. 17) $M = (\text{predecessor}\, M)^+$.

Let M be a non limit aleph. One can check that predecessor M is infinite.

Let X be a set and let N, N_1 be cardinal numbers. An Inf Matrix of N, N_1, X is a function from $[\!: N, N_1 :\!]$ into X.

For simplicity, we follow the rules: X denotes a set, M denotes a non limit aleph, F denotes a filter of M, N_1, N_2 denote elements of predecessor M, K_1, K_2 denote elements of M, and T denotes an Inf Matrix of predecessor M, M, 2^M.

Let us consider M, T. We say that T is Ulam Matrix of M if and only if the conditions (Def. 18) are satisfied.

(Def. 18)(i) For all N_1, K_1, K_2 such that $K_1 \neq K_2$ holds $T(N_1, K_1) \cap T(N_1, K_2)$ is empty,

(ii) for all K_1, N_1, N_2 such that $N_1 \neq N_2$ holds $T(N_1, K_1) \cap T(N_2, K_1)$ is empty,

(iii) for every N_1 holds $\overline{\overline{M \setminus \bigcup\{T(N_1, K_1) : K_1 \in M\}}} \leqslant$ predecessor M, and

(iv) for every K_1 holds $\overline{\overline{M \setminus \bigcup\{T(N_1, K_1) : N_1 \in \text{predecessor}\, M\}}} \leqslant$ predecessor M.

The following four propositions are true:

(34) There exists T such that T is Ulam Matrix of M.

(35) Let given M and I be an ideal of M. Suppose I is complete with M and Frechet_Ideal $M \subseteq I$. Then there exists a subset S of 2^M such that $\overline{\overline{S}} = M$ and for every set X_1 such that $X_1 \in S$ holds $X_1 \notin I$ and for all sets X_1, X_2 such that $X_1 \in S$ and $X_2 \in S$ and $X_1 \neq X_2$ holds $X_1 \cap X_2 = \emptyset$.

(36) For every X and for every cardinal number N such that $N \leqslant \overline{\overline{X}}$ there exists a set Y such that $Y \subseteq X$ and $\overline{\overline{Y}} = N$.

(37) For every M it is not true that there exists F such that F is uniform and an ultrafilter and F is complete with M.

In the sequel M is an aleph.

The following four propositions are true:

(38) If M is measurable, then M is limit.

(39) If M is measurable, then M is inaccessible.

(40) If M is measurable, then M is strong limit.

(41) If M is measurable, then M is strongly inaccessible.

References

[1] Grzegorz Bancerek. Cardinal arithmetics. *Formalized Mathematics*, 1(**3**):543–547, 1990.

[2] Grzegorz Bancerek. Cardinal numbers. *Formalized Mathematics*, 1(**2**):377–382, 1990.

[3] Grzegorz Bancerek. The ordinal numbers. *Formalized Mathematics*, 1(**1**):91–96, 1990.

[4] Grzegorz Bancerek. Sequences of ordinal numbers. *Formalized Mathematics*, 1(**2**):281–290, 1990.

[5] Grzegorz Bancerek. On powers of cardinals. *Formalized Mathematics*, 3(**1**):89–93, 1992.

[6] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[7] Czesław Byliński. Functions from a set to a set. *Formalized Mathematics*, 1(**1**):153–164, 1990.

[8] Czesław Byliński. Some basic properties of sets. *Formalized Mathematics*, 1(**1**):47–53, 1990.

[9] Agata Darmochwał. Finite sets. *Formalized Mathematics*, 1(**1**):165–167, 1990.

[10] Beata Padlewska. Families of sets. *Formalized Mathematics*, 1(**1**):147–152, 1990.

[11] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[12] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[13] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

[14] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

Received April 14, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

The Evaluation of Multivariate Polynomials

Christoph Schwarzweller
University of Tübingen

Andrzej Trybulec
University of Białystok

MML Identifier: POLYNOM2.

The notation and terminology used in this paper are introduced in the following papers: [14], [5], [25], [3], [20], [7], [8], [6], [18], [22], [1], [19], [23], [2], [17], [15], [4], [9], [26], [21], [10], [24], [16], [12], [11], and [13].

1. Preliminaries

In this article we present several logical schemes. The scheme *FinRecExD2* deals with a non empty set $\mathcal{A}$, an element $\mathcal{B}$ of $\mathcal{A}$, a natural number $\mathcal{C}$, and a ternary predicate $\mathcal{P}$, and states that:

> There exists a finite sequence p of elements of $\mathcal{A}$ such that $\operatorname{len} p = \mathcal{C}$ but $p_1 = \mathcal{B}$ or $\mathcal{C} = 0$ but for every natural number n such that $1 \leqslant n$ and $n < \mathcal{C}$ holds $\mathcal{P}[n, p_n, p_{n+1}]$

provided the parameters meet the following conditions:

- Let n be a natural number. Suppose $1 \leqslant n$ and $n < \mathcal{C}$. Let x be an element of $\mathcal{A}$. Then there exists an element y of $\mathcal{A}$ such that $\mathcal{P}[n, x, y]$, and
- Let n be a natural number. Suppose $1 \leqslant n$ and $n < \mathcal{C}$. Let x, y_1, y_2 be elements of $\mathcal{A}$. If $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$, then $y_1 = y_2$.

The scheme *FinRecUnD2* deals with a non empty set $\mathcal{A}$, an element $\mathcal{B}$ of $\mathcal{A}$, a natural number $\mathcal{C}$, finite sequences $\mathcal{D}$, $\mathcal{E}$ of elements of $\mathcal{A}$, and a ternary predicate $\mathcal{P}$, and states that:

> $\mathcal{D} = \mathcal{E}$

provided the parameters meet the following requirements:

- Let n be a natural number. Suppose $1 \leqslant n$ and $n < \mathcal{C}$. Let x, y_1, y_2 be elements of $\mathcal{A}$. If $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$, then $y_1 = y_2$,

- $\operatorname{len}\mathcal{D} = \mathcal{C}$ but $\mathcal{D}_1 = \mathcal{B}$ or $\mathcal{C} = 0$ but for every natural number n such that $1 \leqslant n$ and $n < \mathcal{C}$ holds $\mathcal{P}[n, \mathcal{D}_n, \mathcal{D}_{n+1}]$, and
- $\operatorname{len}\mathcal{E} = \mathcal{C}$ but $\mathcal{E}_1 = \mathcal{B}$ or $\mathcal{C} = 0$ but for every natural number n such that $1 \leqslant n$ and $n < \mathcal{C}$ holds $\mathcal{P}[n, \mathcal{E}_n, \mathcal{E}_{n+1}]$.

The scheme *FinInd* deals with natural numbers $\mathcal{A}$, $\mathcal{B}$ and a unary predicate $\mathcal{P}$, and states that:

For every natural number i such that $\mathcal{A} \leqslant i$ and $i \leqslant \mathcal{B}$ holds $\mathcal{P}[i]$

provided the following conditions are satisfied:

- $\mathcal{P}[\mathcal{A}]$, and
- For every natural number j such that $\mathcal{A} \leqslant j$ and $j < \mathcal{B}$ holds if $\mathcal{P}[j]$, then $\mathcal{P}[j + 1]$.

The scheme *FinInd2* deals with natural numbers $\mathcal{A}$, $\mathcal{B}$ and a unary predicate $\mathcal{P}$, and states that:

For every natural number i such that $\mathcal{A} \leqslant i$ and $i \leqslant \mathcal{B}$ holds $\mathcal{P}[i]$

provided the parameters satisfy the following conditions:

- $\mathcal{P}[\mathcal{A}]$, and
- Let j be a natural number. Suppose $\mathcal{A} \leqslant j$ and $j < \mathcal{B}$. Suppose that for every natural number j' such that $\mathcal{A} \leqslant j'$ and $j' \leqslant j$ holds $\mathcal{P}[j']$. Then $\mathcal{P}[j + 1]$.

The scheme *IndFinSeq* deals with a set $\mathcal{A}$, a finite sequence $\mathcal{B}$ of elements of $\mathcal{A}$, and a unary predicate $\mathcal{P}$, and states that:

For every natural number i such that $1 \leqslant i$ and $i \leqslant \operatorname{len}\mathcal{B}$ holds $\mathcal{P}[\mathcal{B}(i)]$

provided the following conditions are satisfied:

- $\mathcal{P}[\mathcal{B}(1)]$, and
- For every natural number i such that $1 \leqslant i$ and $i < \operatorname{len}\mathcal{B}$ holds if $\mathcal{P}[\mathcal{B}(i)]$, then $\mathcal{P}[\mathcal{B}(i + 1)]$.

Let us mention that every non empty double loop structure which is commutative and right distributive is also distributive.

The following two propositions are true:

(1) Let L be an add-associative right zeroed right complementable distributive non empty double loop structure and x, y be elements of the carrier of L. Then $(-x) \cdot y = -x \cdot y$.

(2) Let L be a unital associative non trivial non empty double loop structure, a be an element of the carrier of L, and n, m be natural numbers. Then $\operatorname{power}_L(a, n + m) = \operatorname{power}_L(a, n) \cdot \operatorname{power}_L(a, m)$.

Let us note that every non empty multiplicative loop structure which is well unital is also unital.

One can prove the following proposition

(3) For every well unital non empty double loop structure L holds $\mathbf{1}_L = 1_L$.

Let us note that there exists a non empty double loop structure which is Abelian, right zeroed, add-associative, right complementable, unital, well unital, distributive, commutative, associative, and non trivial.

2. About Finite Sequences and the Functor SgmX

Next we state a number of propositions:

(4) Let D be a set, p be a finite sequence of elements of D, and k be a natural number. Suppose $k \in \operatorname{dom} p$. Let i be a natural number. If $1 \leqslant i$ and $i \leqslant k$, then $i \in \operatorname{dom} p$.

(5) Let L be a left zeroed right zeroed non empty loop structure, p be a finite sequence of elements of the carrier of L, and i be a natural number. Suppose $i \in \operatorname{dom} p$ and for every natural number i' such that $i' \in \operatorname{dom} p$ and $i' \neq i$ holds $p_{i'} = 0_L$. Then $\sum p = p_i$.

(6) Let L be an add-associative right zeroed right complementable distributive unital non empty double loop structure and p be a finite sequence of elements of the carrier of L. If there exists a natural number i such that $i \in \operatorname{dom} p$ and $p_i = 0_L$, then $\prod p = 0_L$.

(7) Let L be an Abelian add-associative non empty loop structure, a be an element of the carrier of L, and p, q be finite sequences of elements of the carrier of L. Suppose that

(i) $\operatorname{len} p = \operatorname{len} q$, and

(ii) there exists a natural number i such that $i \in \operatorname{dom} p$ and $q_i = a + p_i$ and for every natural number i' such that $i' \in \operatorname{dom} p$ and $i' \neq i$ holds $q_{i'} = p_{i'}$.

Then $\sum q = a + \sum p$.

(8) Let L be a commutative associative non empty double loop structure, a be an element of the carrier of L, and p, q be finite sequences of elements of the carrier of L. Suppose that

(i) $\operatorname{len} p = \operatorname{len} q$, and

(ii) there exists a natural number i such that $i \in \operatorname{dom} p$ and $q_i = a \cdot p_i$ and for every natural number i' such that $i' \in \operatorname{dom} p$ and $i' \neq i$ holds $q_{i'} = p_{i'}$.

Then $\prod q = a \cdot \prod p$.

(9) Let X be a set, A be an empty subset of X, and R be an order in X. If R linearly orders A, then $\operatorname{SgmX}(R, A) = \varepsilon$.

(10) Let X be a set, A be a finite subset of X, and R be an order in X. Suppose R linearly orders A. Let i, j be natural numbers. If $i \in \operatorname{dom} \operatorname{SgmX}(R, A)$ and $j \in \operatorname{dom} \operatorname{SgmX}(R, A)$, then if $(\operatorname{SgmX}(R, A))_i = (\operatorname{SgmX}(R, A))_j$, then $i = j$.

(11) Let X be a set, A be a finite subset of X, and a be an element of X. Suppose $a \notin A$. Let B be a finite subset of X. Suppose $B = \{a\} \cup A$. Let R

be an order in X. Suppose R linearly orders B. Let k be a natural number. Suppose $k \in \operatorname{dom} \operatorname{SgmX}(R, B)$ and $(\operatorname{SgmX}(R, B))_k = a$. Let i be a natural number. If $1 \leqslant i$ and $i \leqslant k - 1$, then $(\operatorname{SgmX}(R, B))_i = (\operatorname{SgmX}(R, A))_i$.

(12) Let X be a set, A be a finite subset of X, and a be an element of X. Suppose $a \notin A$. Let B be a finite subset of X. Suppose $B = \{a\} \cup A$. Let R be an order in X. Suppose R linearly orders B. Let k be a natural number. Suppose $k \in \operatorname{dom} \operatorname{SgmX}(R, B)$ and $(\operatorname{SgmX}(R, B))_k = a$. Let i be a natural number. If $k \leqslant i$ and $i \leqslant \operatorname{len} \operatorname{SgmX}(R, A)$, then $(\operatorname{SgmX}(R, B))_{i+1} = (\operatorname{SgmX}(R, A))_i$.

(13) Let X be a non empty set, A be a finite subset of X, and a be an element of X. Suppose $a \notin A$. Let B be a finite subset of X. Suppose $B = \{a\} \cup A$. Let R be an order in X. Suppose R linearly orders B. Let k be a natural number. If $k + 1 \in \operatorname{dom} \operatorname{SgmX}(R, B)$ and $(\operatorname{SgmX}(R, B))_{k+1} = a$, then $\operatorname{SgmX}(R, B) = \operatorname{Ins}(\operatorname{SgmX}(R, A), k, a)$.

Let n be an ordinal number. Then $\subseteq_n$ is an order in n.

3. Evaluation of Bags

Next we state the proposition

(14) For every set X and for every bag b of X such that $\operatorname{support} b = \emptyset$ holds $b = \operatorname{EmptyBag} X$.

Let X be a set and let b be a bag of X. We say that b is empty if and only if:

(Def. 1) $b = \operatorname{EmptyBag} X$.

Let X be a non empty set. Observe that there exists a bag of X which is non empty.

Let X be a set and let b be a bag of X. Then $\operatorname{support} b$ is a finite subset of X.

Next we state the proposition

(15) For every ordinal number n and for every bag b of n holds $\subseteq_n$ linearly orders $\operatorname{support} b$.

Let X be a set, let x be a finite sequence of elements of X, and let b be a bag of X. Then $b \cdot x$ is a partial function from $\mathbb{N}$ to $\mathbb{N}$.

Let n be an ordinal number, let b be a bag of n, let L be a non trivial unital non empty double loop structure, and let x be a function from n into L. The functor $\operatorname{eval}(b, x)$ yields an element of L and is defined by the condition (Def. 2).

(Def. 2) There exists a finite sequence y of elements of the carrier of L such that

(i) $\operatorname{len} y = \operatorname{len} \operatorname{SgmX}(\subseteq_n, \operatorname{support} b) + 1$,

(ii) $y_1 = 1_L$,

(iii) $\mathrm{eval}(b, x) = \prod y$, and

(iv) for every natural number i such that $1 < i$ and $i \leqslant \mathrm{len}\, y$ holds $y_i = \mathrm{power}_L((x \cdot \mathrm{SgmX}(\subseteq_n, \mathrm{support}\, b))_{i-1},\ (b \cdot \mathrm{SgmX}(\subseteq_n, \mathrm{support}\, b))_{i-1})$.

Next we state three propositions:

(16) Let n be an ordinal number, L be a non trivial unital non empty double loop structure, and x be a function from n into L. Then $\mathrm{eval}(\mathrm{EmptyBag}\, n, x) = 1_L$.

(17) Let n be an ordinal number, L be a unital non trivial non empty double loop structure, u be a set, and b be a bag of n. If $\mathrm{support}\, b = \{u\}$, then for every function x from n into L holds $\mathrm{eval}(b, x) = \mathrm{power}_L(x(u),\ b(u))$.

(18) Let n be an ordinal number, L be a right zeroed add-associative right complementable unital distributive Abelian non trivial commutative associative non empty double loop structure, b_1, b_2 be bags of n, and x be a function from n into L. Then $\mathrm{eval}(b_1 + b_2, x) = \mathrm{eval}(b_1, x) \cdot \mathrm{eval}(b_2, x)$.

4. Evaluation of Polynomials

Let n be an ordinal number, let L be an add-associative right zeroed right complementable non empty loop structure, and let p, q be Polynomials of n, L. Note that $p - q$ is finite-Support.

The following proposition is true

(19) Let L be a right zeroed add-associative right complementable unital distributive non trivial non empty double loop structure, n be an ordinal number, and p be a Polynomial of n, L. If $\mathrm{Support}\, p = \emptyset$, then $p = 0\_(n, L)$.

Let n be an ordinal number, let L be a right zeroed add-associative right complementable unital distributive non trivial non empty double loop structure, and let p be a Polynomial of n, L. Note that $\mathrm{Support}\, p$ is finite.

Next we state the proposition

(20) Let n be an ordinal number, L be a right zeroed add-associative right complementable unital distributive non trivial non empty double loop structure, and p be a Polynomial of n, L. Then $\mathrm{BagOrder}\, n$ linearly orders $\mathrm{Support}\, p$.

Let n be an ordinal number and let b be an element of $\mathrm{Bags}\, n$. The functor b^{T} yields a bag of n and is defined as follows:

(Def. 3) $b^{\mathrm{T}} = b$.

Let n be an ordinal number, let L be a right zeroed add-associative right complementable unital distributive non trivial non empty double loop structure, let p be a Polynomial of n, L, and let x be a function from n into L. The functor $\mathrm{eval}(p, x)$ yields an element of L and is defined by the condition (Def. 4).

(Def. 4) There exists a finite sequence y of elements of the carrier of L such that

(i) $\operatorname{len} y = \operatorname{len} \mathrm{SgmX}(\mathrm{BagOrder}\, n, \mathrm{Support}\, p) + 1$,

(ii) $y_1 = 0_L$,

(iii) $\mathrm{eval}(p, x) = \sum y$, and

(iv) for every natural number i such that $1 < i$ and $i \leqslant \operatorname{len} y$ holds $y_i = (p \cdot \mathrm{SgmX}(\mathrm{BagOrder}\, n, \mathrm{Support}\, p))_{i-1} \cdot \mathrm{eval}(((\mathrm{SgmX}(\mathrm{BagOrder}\, n, \mathrm{Support}\, p))_{i-1})^{\mathrm{T}}, x)$.

One can prove the following propositions:

(21) Let n be an ordinal number, L be a right zeroed add-associative right complementable unital distributive non trivial non empty double loop structure, p be a Polynomial of n, L, and b be a bag of n. If $\mathrm{Support}\, p = \{b\}$, then for every function x from n into L holds $\mathrm{eval}(p, x) = p(b) \cdot \mathrm{eval}(b, x)$.

(22) Let n be an ordinal number, L be a right zeroed add-associative right complementable unital distributive non trivial non empty double loop structure, and x be a function from n into L. Then $\mathrm{eval}(0\_(n, L), x) = 0_L$.

(23) Let n be an ordinal number, L be a right zeroed add-associative right complementable unital distributive non trivial non empty double loop structure, and x be a function from n into L. Then $\mathrm{eval}(1\_(n, L), x) = 1_L$.

(24) Let n be an ordinal number, L be a right zeroed add-associative right complementable unital distributive non trivial non empty double loop structure, p be a Polynomial of n, L, and x be a function from n into L. Then $\mathrm{eval}(-p, x) = -\mathrm{eval}(p, x)$.

(25) Let n be an ordinal number, L be a right zeroed add-associative right complementable Abelian unital distributive non trivial non empty double loop structure, p, q be Polynomials of n, L, and x be a function from n into L. Then $\mathrm{eval}(p + q, x) = \mathrm{eval}(p, x) + \mathrm{eval}(q, x)$.

(26) Let n be an ordinal number, L be a right zeroed add-associative right complementable Abelian unital distributive non trivial non empty double loop structure, p, q be Polynomials of n, L, and x be a function from n into L. Then $\mathrm{eval}(p - q, x) = \mathrm{eval}(p, x) - \mathrm{eval}(q, x)$.

(27) Let n be an ordinal number, L be a right zeroed add-associative right complementable Abelian unital distributive non trivial commutative associative non empty double loop structure, p, q be Polynomials of n, L, and x be a function from n into L. Then $\mathrm{eval}(p * q, x) = \mathrm{eval}(p, x) \cdot \mathrm{eval}(q, x)$.

5. Evaluation Homomorphism

Let n be an ordinal number, let L be a right zeroed add-associative right complementable unital distributive non trivial non empty double loop structure, and let x be a function from n into L. The functor Polynom-Evaluation(n, L, x) yielding a map from Polynom-Ring(n, L) into L is defined by:

(Def. 5) For every Polynomial p of n, L holds (Polynom-Evaluation$(n, L, x))(p) =$ eval(p, x).

Let n be an ordinal number and let L be a right zeroed Abelian add-associative right complementable well unital distributive associative non trivial non empty double loop structure. One can check that Polynom-Ring(n, L) is well unital.

Let n be an ordinal number, let L be an Abelian right zeroed add-associative right complementable well unital distributive associative non trivial non empty double loop structure, and let x be a function from n into L.

Note that Polynom-Evaluation(n, L, x) is unity-preserving.

Let n be an ordinal number, let L be a right zeroed add-associative right complementable Abelian unital distributive non trivial non empty double loop structure, and let x be a function from n into L. One can verify that Polynom-Evaluation(n, L, x) is additive.

Let n be an ordinal number, let L be a right zeroed add-associative right complementable Abelian unital distributive non trivial commutative associative non empty double loop structure, and let x be a function from n into L. Note that Polynom-Evaluation(n, L, x) is multiplicative.

Let n be an ordinal number, let L be a right zeroed add-associative right complementable Abelian well unital distributive non trivial commutative associative non empty double loop structure, and let x be a function from n into L. One can verify that Polynom-Evaluation(n, L, x) is ring homomorphism.

References

[1] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[2] Grzegorz Bancerek. The ordinal numbers. *Formalized Mathematics*, 1(**1**):91–96, 1990.

[3] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[4] Józef Białas. Group and field definitions. *Formalized Mathematics*, 1(**3**):433–439, 1990.

[5] Czesław Byliński. Binary operations. *Formalized Mathematics*, 1(**1**):175–180, 1990.

[6] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[7] Czesław Byliński. Some properties of restrictions of finite sequences. *Formalized Mathematics*, 5(**2**):241–245, 1996.

[8] Agata Darmochwał. Finite sets. *Formalized Mathematics*, 1(**1**):165–167, 1990.

[9] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[10] Jarosław Kotowicz. Monotone real sequences. Subsequences. *Formalized Mathematics*, 1(**3**):471–475, 1990.

[11] Eugeniusz Kusak, Wojciech Leończuk, and Michał Muzalewski. Abelian groups, fields and vector spaces. *Formalized Mathematics*, 1(**2**):335–342, 1990.

[12] Beata Madras. On the concept of the triangulation. *Formalized Mathematics*, 5(**3**):457–462, 1996.

[13] Michał Muzalewski. Construction of rings and left-, right-, and bi-modules over a ring. *Formalized Mathematics*, 2(**1**):3–11, 1991.

[14] Michał Muzalewski and Wojciech Skaba. From loops to abelian multiplicative groups with zero. *Formalized Mathematics*, 1(**5**):833–840, 1990.

[15] Piotr Rudnicki and Andrzej Trybulec. Multivariate polynomials with arbitrary number of variables. *Formalized Mathematics*, 9(**1**):95–110, 2001.

[16] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[17] Andrzej Trybulec. Many-sorted sets. *Formalized Mathematics*, 4(**1**):15–22, 1993.

[18] Wojciech A. Trybulec. Groups. *Formalized Mathematics*, 1(**5**):821–827, 1990.

[19] Wojciech A. Trybulec. Partially ordered sets. *Formalized Mathematics*, 1(**2**):313–319, 1990.

[20] Wojciech A. Trybulec. Pigeon hole principle. *Formalized Mathematics*, 1(**3**):575–579, 1990.

[21] Wojciech A. Trybulec. Vectors in real linear space. *Formalized Mathematics*, 1(**2**):291–296, 1990.

[22] Wojciech A. Trybulec. Lattice of subgroups of a group. Frattini subgroup. *Formalized Mathematics*, 2(**1**):41–47, 1991.

[23] Wojciech A. Trybulec and Grzegorz Bancerek. Kuratowski - Zorn lemma. *Formalized Mathematics*, 1(**2**):387–393, 1990.

[24] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[25] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

[26] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

Received April 14, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

The Ring of Polynomials

Robert Milewski
University of Białystok

MML Identifier: POLYNOM3.

The papers [12], [16], [13], [21], [2], [3], [7], [17], [4], [5], [10], [18], [1], [14], [15], [22], [23], [19], [6], [20], [8], [11], and [9] provide the notation and terminology for this paper.

1. Preliminaries

The following four propositions are true:

(1) Let L be an add-associative right zeroed right complementable non empty loop structure and p be a finite sequence of elements of the carrier of L. If for every natural number i such that $i \in \operatorname{dom} p$ holds $p(i) = 0_L$, then $\sum p = 0_L$.

(2) Let V be an Abelian add-associative right zeroed non empty loop structure and p be a finite sequence of elements of the carrier of V. Then $\sum p = \sum \operatorname{Rev}(p)$.

(3) For every finite sequence p of elements of $\mathbb{R}$ holds $\sum p = \sum \operatorname{Rev}(p)$.

(4) For every finite sequence p of elements of $\mathbb{N}$ and for every natural number i such that $i \in \operatorname{dom} p$ holds $\sum p \geqslant p(i)$.

Let D be a non empty set, let i be a natural number, and let p be a finite sequence of elements of D. Then $p{\restriction}_i$ is a finite sequence of elements of D.

Let D be a non empty set and let a, b be elements of D. Then $\langle a, b \rangle$ is an element of D^2.

Let D be a non empty set, let k, n be natural numbers, let p be an element of D^k, and let q be an element of D^n. Then $p \frown q$ is an element of D^{k+n}.

Let D be a non empty set and let n be a natural number. One can check that every finite sequence of elements of D^n is finite sequence yielding.

Let D be a non empty set, let k, n be natural numbers, let p be a finite sequence of elements of D^k, and let q be a finite sequence of elements of D^n. Then $p \frown q$ is an element of $(D^{k+n})^*$.

In this article we present several logical schemes. The scheme *NonUniqPiSeqExD* deals with a non empty set $\mathcal{A}$, a natural number $\mathcal{B}$, and a binary predicate $\mathcal{P}$, and states that:

> There exists a finite sequence p of elements of $\mathcal{A}$ such that $\operatorname{dom} p = \operatorname{Seg} \mathcal{B}$ and for every natural number k such that $k \in \operatorname{Seg} \mathcal{B}$ holds $\mathcal{P}[k, \pi_k p]$

provided the following condition is satisfied:

- For every natural number k such that $k \in \operatorname{Seg} \mathcal{B}$ there exists an element d of $\mathcal{A}$ such that $\mathcal{P}[k, d]$.

The scheme *SeqOfSeqLambdaD* deals with a non empty set $\mathcal{A}$, a natural number $\mathcal{B}$, a unary functor $\mathcal{F}$ yielding a natural number, and a binary functor $\mathcal{G}$ yielding an element of $\mathcal{A}$, and states that:

> There exists a finite sequence p of elements of $\mathcal{A}^*$ such that
>
> (i) $\operatorname{len} p = \mathcal{B}$, and
>
> (ii) for every natural number k such that $k \in \operatorname{Seg} \mathcal{B}$ holds $\operatorname{len} \pi_k p = \mathcal{F}(k)$ and for every natural number n such that $n \in \operatorname{dom} \pi_k p$ holds $(\pi_k p)(n) = \mathcal{G}(k, n)$

for all values of the parameters.

2. The Lexicographic Order of Finite Sequences

Let n be a natural number and let p, q be elements of $\mathbb{N}^n$. The predicate $p < q$ is defined by the condition (Def. 1).

(Def. 1) There exists a natural number i such that $i \in \operatorname{Seg} n$ and $p(i) < q(i)$ and for every natural number k such that $1 \leqslant k$ and $k < i$ holds $p(k) = q(k)$.

Let us note that the predicate $p < q$ is antisymmetric. We introduce $q > p$ as a synonym of $p < q$.

Let n be a natural number and let p, q be elements of $\mathbb{N}^n$. The predicate $p \leqslant q$ is defined by:

(Def. 2) $p < q$ or $p = q$.

Let us note that the predicate $p \leqslant q$ is reflexive. We introduce $q \geqslant p$ as a synonym of $p \leqslant q$.

We now state three propositions:

(5) Let n be a natural number and p, q, r be elements of $\mathbb{N}^n$. Then

(i) if $p < q$ and $q < r$, then $p < r$, and

(ii) if $p < q$ and $q \leqslant r$ or $p \leqslant q$ and $q < r$ or $p \leqslant q$ and $q \leqslant r$, then $p \leqslant r$.

(6) Let n be a natural number and p, q be elements of $\mathbb{N}^n$. Suppose $p \neq q$. Then there exists a natural number i such that $i \in \operatorname{Seg} n$ and $p(i) \neq q(i)$ and for every natural number k such that $1 \leqslant k$ and $k < i$ holds $p(k) = q(k)$.

(7) For every natural number n and for all elements p, q of $\mathbb{N}^n$ holds $p \leqslant q$ or $p > q$.

Let n be a natural number. The functor TuplesOrder n yielding an order in $\mathbb{N}^n$ is defined by:

(Def. 3) For all elements p, q of $\mathbb{N}^n$ holds $\langle p, q \rangle \in \text{TuplesOrder}\, n$ iff $p \leqslant q$.

Let n be a natural number. Note that TuplesOrder n is linear-order.

3. Decomposition of Natural Numbers

Let i be a non empty natural number and let n be a natural number. The functor $\text{Decomp}(n, i)$ yielding a finite sequence of elements of $\mathbb{N}^i$ is defined by:

(Def. 4) There exists a finite subset A of $\mathbb{N}^i$ such that $\text{Decomp}(n, i) = \text{SgmX}(\text{TuplesOrder}\, i, A)$ and for every element p of $\mathbb{N}^i$ holds $p \in A$ iff $\sum p = n$.

Let i be a non empty natural number and let n be a natural number. Note that $\text{Decomp}(n, i)$ is non empty one-to-one and finite sequence yielding.

The following propositions are true:

(8) For every natural number n holds $\operatorname{len} \text{Decomp}(n, 1) = 1$.

(9) For every natural number n holds $\operatorname{len} \text{Decomp}(n, 2) = n + 1$.

(10) For every natural number n holds $\text{Decomp}(n, 1) = \langle \langle n \rangle \rangle$.

(11) For all natural numbers i, j, n, k_1, k_2 such that $(\text{Decomp}(n, 2))(i) = \langle k_1, n -' k_1 \rangle$ and $(\text{Decomp}(n, 2))(j) = \langle k_2, n -' k_2 \rangle$ holds $i < j$ iff $k_1 < k_2$.

(12) For all natural numbers i, n, k_1, k_2 such that $(\text{Decomp}(n, 2))(i) = \langle k_1, n -' k_1 \rangle$ and $(\text{Decomp}(n, 2))(i + 1) = \langle k_2, n -' k_2 \rangle$ holds $k_2 = k_1 + 1$.

(13) For every natural number n holds $(\text{Decomp}(n, 2))(1) = \langle 0, n \rangle$.

(14) For all natural numbers n, i such that $i \in \operatorname{Seg}(n + 1)$ holds $(\text{Decomp}(n, 2))(i) = \langle i -' 1, (n + 1) -' i \rangle$.

Let L be a non empty groupoid, let p, q, r be sequences of L, and let t be a finite sequence of elements of $\mathbb{N}^3$. The functor $\text{prodTuples}(p, q, r, t)$ yielding an element of (the carrier of L)* is defined by:

(Def. 5) $\operatorname{len} \text{prodTuples}(p, q, r, t) = \operatorname{len} t$ and for every natural number k such that $k \in \operatorname{Seg} \operatorname{len} t$ holds $(\text{prodTuples}(p, q, r, t))(k) = p(\pi_1 \pi_k t) \cdot q(\pi_2 \pi_k t) \cdot r(\pi_3 \pi_k t)$.

One can prove the following propositions:

(15) Let L be a non empty groupoid, p, q, r be sequences of L, t be a finite sequence of elements of $\mathbb{N}^3$, P be a permutation of $\operatorname{dom} t$, and t_1 be a finite sequence of elements of $\mathbb{N}^3$. If $t_1 = t \cdot P$, then $\operatorname{prodTuples}(p, q, r, t_1) = \operatorname{prodTuples}(p, q, r, t) \cdot P$.

(16) For every set D and for every finite sequence f of elements of D^* and for every natural number i holds $\overline{\overline{f{\restriction}i}} = \overline{\overline{f}}{\restriction}i$.

(17) Let p be a finite sequence of elements of $\mathbb{R}$ and q be a finite sequence of elements of $\mathbb{N}$. If $p = q$, then for every natural number i holds $p{\restriction}i = q{\restriction}i$.

(18) For every finite sequence p of elements of $\mathbb{N}$ and for all natural numbers i, j such that $i \leqslant j$ holds $\sum(p{\restriction}i) \leqslant \sum(p{\restriction}j)$.

(19) Let p be a finite sequence of elements of $\mathbb{R}$ and i be a natural number. If $i < \operatorname{len} p$, then $p{\restriction}(i+1) = (p{\restriction}i) \frown \langle p(i+1) \rangle$.

(20) Let p be a finite sequence of elements of $\mathbb{R}$ and i be a natural number. If $i < \operatorname{len} p$, then $\sum(p{\restriction}(i+1)) = \sum(p{\restriction}i) + p(i+1)$.

(21) Let p be a finite sequence of elements of $\mathbb{N}$ and i, j, k_1, k_2 be natural numbers. Suppose $i < \operatorname{len} p$ and $j < \operatorname{len} p$ and $p(i+1) \neq 0$ and $p(j+1) \neq 0$ and $1 \leqslant k_1$ and $1 \leqslant k_2$ and $k_1 \leqslant p(i+1)$ and $k_2 \leqslant p(j+1)$ and $\sum(p{\restriction}i) + k_1 = \sum(p{\restriction}j) + k_2$. Then $i = j$ and $k_1 = k_2$.

(22) Let D_1, D_2 be sets, f_1 be a finite sequence of elements of ${D_1}^*$, f_2 be a finite sequence of elements of ${D_2}^*$, and i_1, i_2, j_1, j_2 be natural numbers. Suppose $i_1 \in \operatorname{dom} f_1$ and $i_2 \in \operatorname{dom} f_2$ and $j_1 \in \operatorname{dom} f_1(i_1)$ and $j_2 \in \operatorname{dom} f_2(i_2)$ and $\overline{\overline{f_1}} = \overline{\overline{f_2}}$ and $\sum(\overline{\overline{f_1}}{\restriction}(i_1 -' 1)) + j_1 = \sum(\overline{\overline{f_2}}{\restriction}(i_2 -' 1)) + j_2$. Then $i_1 = i_2$ and $j_1 = j_2$.

4. Polynomials

Let L be a non empty zero structure. A Polynomial of L is an algebraic sequence of L.

The following proposition is true

(23) Let L be a non empty zero structure, p be a Polynomial of L, and n be a natural number. Then $n \geqslant \operatorname{len} p$ if and only if the length of p is at most n.

Now we present two schemes. The scheme *PolynomialLambda* deals with a non empty loop structure $\mathcal{A}$, a natural number $\mathcal{B}$, and a unary functor $\mathcal{F}$ yielding an element of the carrier of $\mathcal{A}$, and states that:

> There exists a Polynomial p of $\mathcal{A}$ such that $\operatorname{len} p \leqslant \mathcal{B}$ and for every natural number n such that $n < \mathcal{B}$ holds $p(n) = \mathcal{F}(n)$

for all values of the parameters.

The scheme *ExDLoopStrSeq* deals with a non empty loop structure $\mathcal{A}$ and a unary functor $\mathcal{F}$ yielding an element of the carrier of $\mathcal{A}$, and states that:

> There exists a sequence S of $\mathcal{A}$ such that for every natural number n holds $S(n) = \mathcal{F}(n)$

for all values of the parameters.

Let L be a non empty loop structure and let p, q be sequences of L. The functor $p + q$ yielding a sequence of L is defined by:

(Def. 6) For every natural number n holds $(p + q)(n) = p(n) + q(n)$.

Let L be a right zeroed non empty loop structure and let p, q be Polynomials of L. Note that $p + q$ is finite-Support.

One can prove the following two propositions:

(24) Let L be a right zeroed non empty loop structure, p, q be Polynomials of L, and n be a natural number. Suppose the length of p is at most n and the length of q is at most n. Then the length of $p + q$ is at most n.

(25) For every right zeroed non empty loop structure L and for all Polynomials p, q of L holds $\operatorname{support}(p + q) \subseteq \operatorname{support} p \cup \operatorname{support} q$.

Let L be an Abelian non empty loop structure and let p, q be sequences of L. Let us note that the functor $p + q$ is commutative.

One can prove the following proposition

(26) For every add-associative non empty loop structure L and for all sequences p, q, r of L holds $(p + q) + r = p + (q + r)$.

Let L be a non empty loop structure and let p be a sequence of L. The functor $-p$ yielding a sequence of L is defined by:

(Def. 7) For every natural number n holds $(-p)(n) = -p(n)$.

Let L be an add-associative right zeroed right complementable non empty loop structure and let p be a Polynomial of L. Observe that $-p$ is finite-Support.

Let L be a non empty loop structure and let p, q be sequences of L. The functor $p - q$ yields a sequence of L and is defined as follows:

(Def. 8) $p - q = p + -q$.

Let L be an add-associative right zeroed right complementable non empty loop structure and let p, q be Polynomials of L. Note that $p-q$ is finite-Support.

Next we state the proposition

(27) Let L be a non empty loop structure, p, q be sequences of L, and n be a natural number. Then $(p - q)(n) = p(n) - q(n)$.

Let L be a non empty zero structure. The functor $\mathbf{0}.\,L$ yielding a sequence of L is defined as follows:

(Def. 9) $\mathbf{0}.\,L = \mathbb{N} \longmapsto 0_L$.

Let L be a non empty zero structure. One can check that $\mathbf{0}.\,L$ is finite-Support.

We now state three propositions:

(28) For every non empty zero structure L and for every natural number n holds $(\mathbf{0}.\,L)(n) = 0_L$.

(29) For every right zeroed non empty loop structure L and for every sequence p of L holds $p + \mathbf{0}.\,L = p$.

(30) Let L be an add-associative right zeroed right complementable non empty loop structure and p be a sequence of L. Then $p - p = \mathbf{0}.\,L$.

Let L be a non empty multiplicative loop with zero structure. The functor $\mathbf{1}.\,L$ yielding a sequence of L is defined by:

(Def. 10) $\mathbf{1}.\,L = \mathbf{0}.\,L +\cdot\, (0, \mathbf{1}_L)$.

Let L be a non empty multiplicative loop with zero structure. Observe that $\mathbf{1}.\,L$ is finite-Support.

Next we state the proposition

(31) Let L be a non empty multiplicative loop with zero structure. Then $(\mathbf{1}.\,L)(0) = \mathbf{1}_L$ and for every natural number n such that $n \neq 0$ holds $(\mathbf{1}.\,L)(n) = 0_L$.

Let L be a non empty double loop structure and let p, q be sequences of L. The functor $p*q$ yields a sequence of L and is defined by the condition (Def. 11).

(Def. 11) Let i be a natural number. Then there exists a finite sequence r of elements of the carrier of L such that $\operatorname{len} r = i + 1$ and $(p * q)(i) = \sum r$ and for every natural number k such that $k \in \operatorname{dom} r$ holds $r(k) = p(k -' 1) \cdot q((i + 1) -' k)$.

Let L be an add-associative right zeroed right complementable distributive non empty double loop structure and let p, q be Polynomials of L. Note that $p * q$ is finite-Support.

Next we state three propositions:

(32) Let L be an Abelian add-associative right zeroed right complementable right distributive non empty double loop structure and p, q, r be sequences of L. Then $p * (q + r) = p * q + p * r$.

(33) Let L be an Abelian add-associative right zeroed right complementable left distributive non empty double loop structure and p, q, r be sequences of L. Then $(p + q) * r = p * r + q * r$.

(34) Let L be an Abelian add-associative right zeroed right complementable unital associative distributive non empty double loop structure and p, q, r be sequences of L. Then $(p * q) * r = p * (q * r)$.

Let L be an Abelian add-associative right zeroed commutative non empty double loop structure and let p, q be sequences of L. Let us observe that the functor $p * q$ is commutative.

We now state two propositions:

(35) Let L be an add-associative right zeroed right complementable right distributive non empty double loop structure and p be a sequence of L.

Then $p * \mathbf{0}.\,L = \mathbf{0}.\,L$.

(36) Let L be an add-associative right zeroed right unital right complementable right distributive non empty double loop structure and p be a sequence of L. Then $p * \mathbf{1}.\,L = p$.

5. The Ring of Polynomials

Let L be an add-associative right zeroed right complementable distributive non empty double loop structure. The functor Polynom-Ring L yields a strict non empty double loop structure and is defined by the conditions (Def. 12).

(Def. 12)(i) For every set x holds $x \in$ the carrier of Polynom-Ring L iff x is a Polynomial of L,

(ii) for all elements x, y of the carrier of Polynom-Ring L and for all sequences p, q of L such that $x = p$ and $y = q$ holds $x + y = p + q$,

(iii) for all elements x, y of the carrier of Polynom-Ring L and for all sequences p, q of L such that $x = p$ and $y = q$ holds $x \cdot y = p * q$,

(iv) $0_{\text{Polynom-Ring}\,L} = \mathbf{0}.\,L$, and

(v) $\mathbf{1}_{\text{Polynom-Ring}\,L} = \mathbf{1}.\,L$.

Let L be an Abelian add-associative right zeroed right complementable distributive non empty double loop structure. Observe that Polynom-Ring L is Abelian.

Let L be an add-associative right zeroed right complementable distributive non empty double loop structure. One can check the following observations:

$*$ Polynom-Ring L is add-associative,

$*$ Polynom-Ring L is right zeroed, and

$*$ Polynom-Ring L is right complementable.

Let L be an Abelian add-associative right zeroed right complementable commutative distributive non empty double loop structure. Note that Polynom-Ring L is commutative.

Let L be an Abelian add-associative right zeroed right complementable unital associative distributive non empty double loop structure. Observe that Polynom-Ring L is associative.

Let L be an add-associative right zeroed right complementable right unital distributive non empty double loop structure. Observe that Polynom-Ring L is right unital.

Let L be an Abelian add-associative right zeroed right complementable distributive non empty double loop structure. Note that Polynom-Ring L is right distributive and Polynom-Ring L is left distributive.

References

[1] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[2] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[3] Czesław Byliński. Finite sequences and tuples of elements of a non-empty sets. *Formalized Mathematics*, 1(**3**):529–536, 1990.

[4] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[5] Czesław Byliński. Functions from a set to a set. *Formalized Mathematics*, 1(**1**):153–164, 1990.

[6] Czesław Byliński. The sum and product of finite sequences of real numbers. *Formalized Mathematics*, 1(**4**):661–668, 1990.

[7] Czesław Byliński. Some properties of restrictions of finite sequences. *Formalized Mathematics*, 5(**2**):241–245, 1996.

[8] Agata Darmochwał and Yatsuka Nakamura. The topological space $\mathcal{E}^2_{\rm T}$. Arcs, line segments and special polygonal arcs. *Formalized Mathematics*, 2(**5**):617–621, 1991.

[9] Andrzej Kondracki. The Chinese Remainder Theorem. *Formalized Mathematics*, 6(**4**):573–577, 1997.

[10] Jarosław Kotowicz and Yatsuka Nakamura. Introduction to Go-board - part I. *Formalized Mathematics*, 3(**1**):107–115, 1992.

[11] Eugeniusz Kusak, Wojciech Leończuk, and Michał Muzalewski. Abelian groups, fields and vector spaces. *Formalized Mathematics*, 1(**2**):335–342, 1990.

[12] Michał Muzalewski and Lesław W. Szczerba. Construction of finite sequences over ring and left-, right-, and bi-modules over a ring. *Formalized Mathematics*, 2(**1**):97–104, 1991.

[13] Takaya Nishiyama and Yasuho Mizuhara. Binary arithmetics. *Formalized Mathematics*, 4(**1**):83–86, 1993.

[14] Jan Popiołek. Real normed space. *Formalized Mathematics*, 2(**1**):111–115, 1991.

[15] Piotr Rudnicki and Andrzej Trybulec. Multivariate polynomials with arbitrary number of variables. *Formalized Mathematics*, 9(**1**):95–110, 2001.

[16] Wojciech Skaba and Michał Muzalewski. From double loops to fields. *Formalized Mathematics*, 2(**1**):185–191, 1991.

[17] Wojciech A. Trybulec. Binary operations on finite sequences. *Formalized Mathematics*, 1(**5**):979–981, 1990.

[18] Wojciech A. Trybulec. Groups. *Formalized Mathematics*, 1(**5**):821–827, 1990.

[19] Wojciech A. Trybulec. Vectors in real linear space. *Formalized Mathematics*, 1(**2**):291–296, 1990.

[20] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[21] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

[22] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

[23] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

Received April 17, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Solving Roots of Polynomial Equations of Degree 2 and 3 with Real Coefficients

Liang Xiquan
Northeast Normal University
China

Summary. In this paper, we describe the definition of the first, second, and third degree algebraic equations and their properties. In Section 1, we defined the simple first-degree and second-degree (quadratic) equation and discussed the relation between the roots of each equation and their coefficients. Also, we clarified the form of the root within the range of real numbers. Furthermore, the extraction of the root using the discriminant of equation is clarified. In Section 2, we defined the third-degree (cubic) equation and clarified the relation between the three roots of this equation and its coefficient. Also, the form of these roots for various conditions is discussed. This solution is known as the Cardano solution.

MML Identifier: POLYEQ_1.

The terminology and notation used in this paper are introduced in the following articles: [4], [3], [2], [1], [5], and [6].

1. Equation of Degree 1 and 2

Let a, b, x be real numbers. The functor $\mathrm{Poly1}(a, b, x)$ yields a real number and is defined as follows:

(Def. 1) $\mathrm{Poly1}(a, b, x) = a \cdot x + b$.

One can prove the following three propositions:

(1) For all real numbers a, b, x such that $a \neq 0$ holds if $\mathrm{Poly1}(a, b, x) = 0$, then $x = -\frac{b}{a}$.

(2) For every real number x holds $\mathrm{Poly1}(0, 0, x) = 0$.

ISSN 1426–2630

(3) For all real numbers a, b, x such that $a = 0$ and $b \neq 0$ it is not true that there exists a real number x such that $\mathrm{Poly1}(a, b, x) = 0$.

Let a, b, c, x be real numbers. The functor $\mathrm{Poly2}(a, b, c, x)$ yields a real number and is defined by:

(Def. 2) $\mathrm{Poly2}(a, b, c, x) = a \cdot x^{\mathbf{2}} + b \cdot x + c$.

One can prove the following propositions:

(4) For all real numbers a, b, c, a', b', c' such that for every real number x holds $\mathrm{Poly2}(a, b, c, x) = \mathrm{Poly2}(a', b', c', x)$ holds $a = a'$ and $b = b'$ and $c = c'$.

(5) Let a, b, c be real numbers. Suppose $a \neq 0$ and $\Delta(a, b, c) \geqslant 0$. Let x be a real number. If $\mathrm{Poly2}(a, b, c, x) = 0$, then $x = \frac{-b+\sqrt{\Delta(a,b,c)}}{2 \cdot a}$ or $x = \frac{-b-\sqrt{\Delta(a,b,c)}}{2 \cdot a}$.

(6) For all real numbers a, b, c, x such that $a \neq 0$ and $\Delta(a, b, c) = 0$ holds if $\mathrm{Poly2}(a, b, c, x) = 0$, then $x = -\frac{b}{2 \cdot a}$.

(7) For all real numbers a, b, c such that $a \neq 0$ and $\Delta(a, b, c) < 0$ it is not true that there exists a real number x such that $\mathrm{Poly2}(a, b, c, x) = 0$.

(8) For all real numbers a, b, c, x such that $a = 0$ and $b \neq 0$ holds if for every real number x holds $\mathrm{Poly2}(a, b, c, x) = 0$, then $x = -\frac{c}{b}$.

(9) For all real numbers a, b, c, x such that $a = 0$ and $b = 0$ and $c = 0$ holds $\mathrm{Poly2}(a, b, c, x) = 0$.

(10) For all real numbers a, b, c such that $a = 0$ and $b = 0$ and $c \neq 0$ it is not true that there exists a real number x such that $\mathrm{Poly2}(a, b, c, x) = 0$.

Let a, x, x_1, x_2 be real numbers. The functor $\mathrm{Quard}(a, x_1, x_2, x)$ yielding a real number is defined by:

(Def. 3) $\mathrm{Quard}(a, x_1, x_2, x) = a \cdot ((x - x_1) \cdot (x - x_2))$.

Next we state the proposition

(11) Let a, b, c, x, x_1, x_2 be real numbers. Suppose $a \neq 0$. Suppose that for every real number x holds $\mathrm{Poly2}(a, b, c, x) = \mathrm{Quard}(a, x_1, x_2, x)$. Then $\frac{b}{a} = -(x_1 + x_2)$ and $\frac{c}{a} = x_1 \cdot x_2$.

2. Equation of Degree 3

Let a, b, c, d, x be real numbers. The functor $\mathrm{Poly3}(a, b, c, d, x)$ yielding a real number is defined as follows:

(Def. 4) $\mathrm{Poly3}(a, b, c, d, x) = a \cdot x^{3}_{\mathbb{N}} + b \cdot x^{\mathbf{2}} + c \cdot x + d$.

Next we state the proposition

(12) Let a, b, c, d, a', b', c', d' be real numbers. Suppose that for every real number x holds $\mathrm{Poly3}(a,b,c,d,x) = \mathrm{Poly3}(a',b',c',d',x)$. Then $a = a'$ and $b = b'$ and $c = c'$ and $d = d'$.

Let a, x, x_1, x_2, x_3 be real numbers. The functor $\mathrm{Tri}(a,x_1,x_2,x_3,x)$ yields a real number and is defined as follows:

(Def. 5) $\mathrm{Tri}(a,x_1,x_2,x_3,x) = a\cdot((x-x_1)\cdot(x-x_2)\cdot(x-x_3))$.

One can prove the following propositions:

(13) Let a, b, c, d, x, x_1, x_2, x_3 be real numbers. Suppose $a \neq 0$. Suppose that for every real number x holds $\mathrm{Poly3}(a,b,c,d,x) = \mathrm{Tri}(a,x_1,x_2,x_3,x)$. Then $\frac{b}{a} = -(x_1+x_2+x_3)$ and $\frac{c}{a} = x_1\cdot x_2 + x_2\cdot x_3 + x_1\cdot x_3$ and $\frac{d}{a} = -x_1\cdot x_2\cdot x_3$.

(14) For all real numbers y, h holds $(y+h)^3_{\mathbb{N}} = (y^3_{\mathbb{N}})+(3\cdot h\cdot y^{\mathbf{2}}+3\cdot h^{\mathbf{2}}\cdot y)+h^3_{\mathbb{N}}$.

(15) Let a, b, c, d, x be real numbers. Suppose $a \neq 0$. Suppose $\mathrm{Poly3}(a,b,c,d,x) = 0$. Let a_1, a_2, a_3, h, y be real numbers. Suppose $y = x+\frac{b}{3\cdot a}$ and $h = -\frac{b}{3\cdot a}$ and $a_1 = \frac{b}{a}$ and $a_2 = \frac{c}{a}$ and $a_3 = \frac{d}{a}$. Then $(y^3_{\mathbb{N}})+((3\cdot h+a_1)\cdot y^{\mathbf{2}}+(3\cdot h^{\mathbf{2}}+2\cdot(a_1\cdot h)+a_2)\cdot y)+((h^3_{\mathbb{N}})+a_1\cdot h^{\mathbf{2}}+(a_2\cdot h+a_3)) = 0$.

(16) Let a, b, c, d, x be real numbers. Suppose $a \neq 0$. Suppose $\mathrm{Poly3}(a,b,c,d,x) = 0$. Let a_1, a_2, a_3, h, y be real numbers. Suppose $y = x+\frac{b}{3\cdot a}$ and $h = -\frac{b}{3\cdot a}$ and $a_1 = \frac{b}{a}$ and $a_2 = \frac{c}{a}$ and $a_3 = \frac{d}{a}$. Then $(y^3_{\mathbb{N}})+0\cdot y^{\mathbf{2}}+\frac{3\cdot a\cdot c-b^{\mathbf{2}}}{3\cdot a^{\mathbf{2}}}\cdot y+(2\cdot(\frac{b}{3\cdot a})^3_{\mathbb{N}}+\frac{3\cdot a\cdot d-b\cdot c}{3\cdot a^{\mathbf{2}}}) = 0$.

(17) Let a, b, c, d, y be real numbers. Suppose $a \neq 0$. Suppose $(y^3_{\mathbb{N}})+0\cdot y^{\mathbf{2}}+\frac{3\cdot a\cdot c-b^{\mathbf{2}}}{3\cdot a^{\mathbf{2}}}\cdot y+(2\cdot(\frac{b}{3\cdot a})^3_{\mathbb{N}}+\frac{3\cdot a\cdot d-b\cdot c}{3\cdot a^{\mathbf{2}}}) = 0$. Let p, q be real numbers. If $p = \frac{3\cdot a\cdot c-b^{\mathbf{2}}}{3\cdot a^{\mathbf{2}}}$ and $q = 2\cdot(\frac{b}{3\cdot a})^3_{\mathbb{N}}+\frac{3\cdot a\cdot d-b\cdot c}{3\cdot a^{\mathbf{2}}}$, then $\mathrm{Poly3}(1,0,p,q,y) = 0$.

(18) Let p, q, y be real numbers. Suppose $\mathrm{Poly3}(1,0,p,q,y) = 0$. Let u, v be real numbers. If $y = u+v$ and $3\cdot v\cdot u+p = 0$, then $(u^3_{\mathbb{N}})+v^3_{\mathbb{N}} = -q$ and $(u^3_{\mathbb{N}})\cdot v^3_{\mathbb{N}} = (-\frac{p}{3})^3_{\mathbb{N}}$.

(19) Let p, q, y be real numbers. Suppose $\mathrm{Poly3}(1,0,p,q,y) = 0$. Let u, v be real numbers. Suppose $y = u+v$ and $3\cdot v\cdot u+p = 0$. Then

(i) $y = \sqrt[3]{-\frac{q}{2}+\sqrt{\frac{q^{\mathbf{2}}}{4}+(\frac{p}{3})^3_{\mathbb{N}}}}+\sqrt[3]{-\frac{q}{2}-\sqrt{\frac{q^{\mathbf{2}}}{4}+(\frac{p}{3})^3_{\mathbb{N}}}}$, or

(ii) $y = \sqrt[3]{-\frac{q}{2}+\sqrt{\frac{q^{\mathbf{2}}}{4}+(\frac{p}{3})^3_{\mathbb{N}}}}+\sqrt[3]{-\frac{q}{2}+\sqrt{\frac{q^{\mathbf{2}}}{4}+(\frac{p}{3})^3_{\mathbb{N}}}}$, or

(iii) $y = \sqrt[3]{-\frac{q}{2}-\sqrt{\frac{q^{\mathbf{2}}}{4}+(\frac{p}{3})^3_{\mathbb{N}}}}+\sqrt[3]{-\frac{q}{2}-\sqrt{\frac{q^{\mathbf{2}}}{4}+(\frac{p}{3})^3_{\mathbb{N}}}}$.

(20) Let a, b, c, d, x be real numbers. Suppose $a = 0$ and $b \neq 0$ and $\Delta(b,c,d) > 0$. If $\mathrm{Poly3}(a,b,c,d,x) = 0$, then $x = \frac{-c+\sqrt{\Delta(b,c,d)}}{2\cdot b}$ or $x = \frac{-c-\sqrt{\Delta(b,c,d)}}{2\cdot b}$.

(21) Let a, b, c, d, p, q, x be real numbers. Suppose $a \neq 0$ and $b = 0$ and $p = \frac{c}{a}$ and $q = \frac{d}{a}$. Suppose $\mathrm{Poly3}(a,b,c,d,x) = 0$. Let u, v be real numbers. Suppose $x = u+v$ and $3\cdot v\cdot u+p = 0$. Then

(i) $x = \sqrt[3]{-\frac{d}{2\cdot a} + \sqrt{\frac{d^{\mathbf{2}}}{4\cdot a^{\mathbf{2}}} + (\frac{c}{3\cdot a})^{3}_{\mathbb{N}}}} + \sqrt[3]{-\frac{d}{2\cdot a} - \sqrt{\frac{d^{\mathbf{2}}}{4\cdot a^{\mathbf{2}}} + (\frac{c}{3\cdot a})^{3}_{\mathbb{N}}}}$, or

(ii) $x = \sqrt[3]{-\frac{d}{2\cdot a} + \sqrt{\frac{d^{\mathbf{2}}}{4\cdot a^{\mathbf{2}}} + (\frac{c}{3\cdot a})^{3}_{\mathbb{N}}}} + \sqrt[3]{-\frac{d}{2\cdot a} + \sqrt{\frac{d^{\mathbf{2}}}{4\cdot a^{\mathbf{2}}} + (\frac{c}{3\cdot a})^{3}_{\mathbb{N}}}}$, or

(iii) $x = \sqrt[3]{-\frac{d}{2\cdot a} - \sqrt{\frac{d^{\mathbf{2}}}{4\cdot a^{\mathbf{2}}} + (\frac{c}{3\cdot a})^{3}_{\mathbb{N}}}} + \sqrt[3]{-\frac{d}{2\cdot a} - \sqrt{\frac{d^{\mathbf{2}}}{4\cdot a^{\mathbf{2}}} + (\frac{c}{3\cdot a})^{3}_{\mathbb{N}}}}$.

(22) Let a, b, c, d, x be real numbers. Suppose $a \neq 0$ and $\Delta(a,b,c) \geqslant 0$ and $d = 0$. If $\mathrm{Poly3}(a,b,c,d,x) = 0$, then $x = 0$ or $x = \frac{-b+\sqrt{\Delta(a,b,c)}}{2\cdot a}$ or $x = \frac{-b-\sqrt{\Delta(a,b,c)}}{2\cdot a}$.

(23) Let a, b, c, d, x be real numbers. Suppose $a \neq 0$ and $b = 0$ and $\frac{c}{a} < 0$ and $d = 0$. If $\mathrm{Poly3}(a,b,c,d,x) = 0$, then $x = 0$ or $x = \sqrt{-\frac{c}{a}}$ or $x = -\sqrt{-\frac{c}{a}}$.

(24) Let a, b, c, d, x be real numbers. Suppose $a \neq 0$ and $c = 0$. Suppose $\mathrm{Poly3}(a,b,c,d,x) = 0$. Let h be a real number. Suppose $a \cdot x + b = h$ and $h \neq 0$ and $\frac{d}{h} < 0$. Then $x = \frac{h-b}{a}$ or $x = \sqrt{-\frac{d}{h}}$ or $x = -\sqrt{-\frac{d}{h}}$.

References

[1] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[2] Jan Popiołek. Quadratic inequalities. *Formalized Mathematics*, 2(**4**):507–509, 1991.

[3] Konrad Raczkowski. Integer and rational exponents. *Formalized Mathematics*, 2(**1**):125–130, 1991.

[4] Konrad Raczkowski and Andrzej Nędzusiak. Real exponents and logarithms. *Formalized Mathematics*, 2(**2**):213–216, 1991.

[5] Andrzej Trybulec and Czesław Byliński. Some properties of real numbers. *Formalized Mathematics*, 1(**3**):445–449, 1990.

[6] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

Received May 18, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

The Concept of Fuzzy Set and Membership Function and Basic Properties of Fuzzy Set Operation

Takashi Mitsuishi
Shinshu University
Nagano

Noboru Endou
Shinshu University
Nagano

Yasunari Shidama
Shinshu University
Nagano

Summary. This article introduces the fuzzy theory. At first, the definition of fuzzy set characterized by membership function is described. Next, definitions of empty fuzzy set and universal fuzzy set and basic operations of these fuzzy sets are shown. At last, exclusive sum and absolute difference which are special operation are introduced.

MML Identifier: FUZZY_1.

The terminology and notation used in this paper have been introduced in the following articles: [8], [1], [5], [9], [3], [4], [6], [7], and [2].

1. Definition of Membership Function and Fuzzy Set

In this paper C is a non empty set and c is an element of C.

We now state the proposition

(1) $\mathrm{rng}(\chi_{C,C}) \subseteq [0,1]$.

Let us consider C. A partial function from C to $\mathbb{R}$ is said to be a membership function of C if:

(Def. 1) $\mathrm{dom}\,\mathrm{it} = C$ and $\mathrm{rng}\,\mathrm{it} \subseteq [0,1]$.

The following proposition is true

(2) $\chi_{C,C}$ is a membership function of C.

ISSN 1426–2630

In the sequel f, h, g, h_1 denote membership functions of C.

Let C be a non empty set and let h be a membership function of C. A set is called a FuzzySet of C, h if:

(Def. 2) It $= [\!: C,\ h^\circ C\, :\!]$.

Let C be a non empty set, let h, g be membership functions of C, let A be a FuzzySet of C, h, and let B be a FuzzySet of C, g. The predicate $A = B$ is defined as follows:

(Def. 3) For every element c of C holds $h(c) = g(c)$.

Let C be a non empty set, let h, g be membership functions of C, let A be a FuzzySet of C, h, and let B be a FuzzySet of C, g. The predicate $A \subseteq B$ is defined by:

(Def. 4) For every element c of C holds $h(c) \leqslant g(c)$.

In the sequel A denotes a FuzzySet of C, f, B denotes a FuzzySet of C, g, D denotes a FuzzySet of C, h, and D_1 denotes a FuzzySet of C, h_1.

One can prove the following propositions:

(3) $A = B$ iff $A \subseteq B$ and $B \subseteq A$.

(4) $A \subseteq A$.

(5) If $A \subseteq B$ and $B \subseteq D$, then $A \subseteq D$.

2. Intersection, Union and Complement

Let C be a non empty set and let h, g be membership functions of C. The functor $\min(h, g)$ yielding a membership function of C is defined by:

(Def. 5) For every element c of C holds $(\min(h, g))(c) = \min(h(c), g(c))$.

Let C be a non empty set and let h, g be membership functions of C. The functor $\max(h, g)$ yields a membership function of C and is defined by:

(Def. 6) For every element c of C holds $(\max(h, g))(c) = \max(h(c), g(c))$.

Let C be a non empty set and let h be a membership function of C. The functor 1-minus h yielding a membership function of C is defined by:

(Def. 7) For every element c of C holds $(\text{1-minus}\, h)(c) = 1 - h(c)$.

Let C be a non empty set, let h, g be membership functions of C, let A be a FuzzySet of C, h, and let B be a FuzzySet of C, g. The functor $A \cap B$ yielding a FuzzySet of C, $\min(h, g)$ is defined as follows:

(Def. 8) $A \cap B = [\!: C,\ (\min(h, g))^\circ C\, :\!]$.

Let C be a non empty set, let h, g be membership functions of C, let A be a FuzzySet of C, h, and let B be a FuzzySet of C, g. The functor $A \cup B$ yields a FuzzySet of C, $\max(h, g)$ and is defined by:

(Def. 9) $A \cup B = [\!: C,\ (\max(h, g))^\circ C\, :\!]$.

Let C be a non empty set, let h be a membership function of C, and let A be a FuzzySet of C, h. The functor $A^{\rm c}$ yielding a FuzzySet of C, 1-minus h is defined by:

(Def. 10) $A^{\rm c} = [\!: C, (\text{1-minus}\, h)^{\circ}C :\!]$.

We now state a number of propositions:

(6) $\min(h(c), g(c)) = (\min(h, g))(c)$ and $\max(h(c), g(c)) = (\max(h, g))(c)$.

(7) $\max(h, h) = h$ and $\min(h, h) = h$ and $\max(h, h) = \min(h, h)$ and $\min(f, g) = \min(g, f)$ and $\max(f, g) = \max(g, f)$.

(8) $f = g$ iff $A = B$.

(9) $A \cap A = A$ and $A \cup A = A$.

(10) $A \cap B = B \cap A$ and $A \cup B = B \cup A$.

(11) $\max(\max(f, g), h) = \max(f, \max(g, h))$ and $\min(\min(f, g), h) = \min(f, \min(g, h))$.

(12) $(A \cup B) \cup D = A \cup (B \cup D)$.

(13) $(A \cap B) \cap D = A \cap (B \cap D)$.

(14) $\max(f, \min(f, g)) = f$ and $\min(f, \max(f, g)) = f$.

(15) $A \cup A \cap B = A$ and $A \cap (A \cup B) = A$.

(16) $\min(f, \max(g, h)) = \max(\min(f, g), \min(f, h))$ and $\max(f, \min(g, h)) = \min(\max(f, g), \max(f, h))$.

(17) $A \cup B \cap D = (A \cup B) \cap (A \cup D)$ and $A \cap (B \cup D) = A \cap B \cup A \cap D$.

(18) 1-minus 1-minus $h = h$.

(19) $(A^{\rm c})^{\rm c} = A$.

(20) 1-minus $\max(f, g) = \min(\text{1-minus}\, f, \text{1-minus}\, g)$ and 1-minus $\min(f, g) = \max(\text{1-minus}\, f, \text{1-minus}\, g)$.

(21) $(A \cup B)^{\rm c} = A^{\rm c} \cap B^{\rm c}$ and $(A \cap B)^{\rm c} = A^{\rm c} \cup B^{\rm c}$.

3. Empty Fuzzy Set and Universal Fuzzy Set

Let C be a non empty set. A set is called an Empty FuzzySet of C if:

(Def. 11) It $= [\!: C, (\chi_{\emptyset,C})^{\circ}C :\!]$.

Let C be a non empty set. A set is called a Universal FuzzySet of C if:

(Def. 12) It $= [\!: C, (\chi_{C,C})^{\circ}C :\!]$.

In the sequel X is a Universal FuzzySet of C and E is an Empty FuzzySet of C.

One can prove the following two propositions:

(22) $\operatorname{rng}(\chi_{\emptyset,C}) \subseteq [0, 1]$.

(23) $\chi_{\emptyset,C}$ is a membership function of C.

Let C be a non empty set. The functor $\mathrm{EMF}\, C$ yields a membership function of C and is defined as follows:

(Def. 13) $\mathrm{EMF}\, C = \chi_{\emptyset,C}$.

Let C be a non empty set. The functor $\mathrm{UMF}\, C$ yields a membership function of C and is defined as follows:

(Def. 14) $\mathrm{UMF}\, C = \chi_{C,C}$.

One can prove the following propositions:

(24) For every membership function h of C such that $h = \chi_{C,C}$ holds $[\!: C, (\chi_{C,C})^{\circ} C :\!]$ is a FuzzySet of C, h.

(25) For every membership function h of C such that $h = \chi_{\emptyset,C}$ holds $[\!: C, (\chi_{\emptyset,C})^{\circ} C :\!]$ is a FuzzySet of C, h.

(26) E is a FuzzySet of C, $\mathrm{EMF}\, C$.

(27) X is a FuzzySet of C, $\mathrm{UMF}\, C$.

Let C be a non empty set. We see that the Empty FuzzySet of C is a FuzzySet of C, $\mathrm{EMF}\, C$.

Let C be a non empty set. We see that the Universal FuzzySet of C is a FuzzySet of C, $\mathrm{UMF}\, C$.

In the sequel X denotes a Universal FuzzySet of C and E denotes an Empty FuzzySet of C.

One can prove the following propositions:

(28) Let a, b be elements of $\mathbb{R}$ and f be a partial function from C to $\mathbb{R}$. Suppose $\mathrm{rng}\, f \subseteq [a, b]$ and $\mathrm{dom}\, f \neq \emptyset$ and $a \leqslant b$. Let x be an element of C. If $x \in \mathrm{dom}\, f$, then $a \leqslant f(x)$ and $f(x) \leqslant b$.

(29) $E \subseteq A$.

(30) $A \subseteq X$.

(31) For every element x of C and for every membership function h of C holds $(\mathrm{EMF}\, C)(x) \leqslant h(x)$ and $h(x) \leqslant (\mathrm{UMF}\, C)(x)$.

(32) $\max(f, \mathrm{UMF}\, C) = \mathrm{UMF}\, C$ and $\min(f, \mathrm{UMF}\, C) = f$ and $\max(f, \mathrm{EMF}\, C) = f$ and $\min(f, \mathrm{EMF}\, C) = \mathrm{EMF}\, C$.

(33) $A \cup X = X$ and $A \cap X = A$.

(34) $A \cup E = A$ and $A \cap E = E$.

(35) $A \subseteq A \cup B$.

(36) If $A \subseteq D$ and $B \subseteq D$, then $A \cup B \subseteq D$.

(37) For all elements a, b, c of $\mathbb{R}$ such that $a \leqslant b$ holds $\max(a, c) \leqslant \max(b, c)$.

(38) If $A \subseteq B$, then $A \cup D \subseteq B \cup D$.

(39) If $A \subseteq B$ and $D \subseteq D_1$, then $A \cup D \subseteq B \cup D_1$.

(40) If $A \subseteq B$, then $A \cup B = B$.

(41) $A \cap B \subseteq A$.

(42) $A \cap B \subseteq A \cup B$.

(43) If $D \subseteq A$ and $D \subseteq B$, then $D \subseteq A \cap B$.

(44) For all elements a, b, c, d of $\mathbb{R}$ such that $a \leqslant b$ and $c \leqslant d$ holds $\min(a, c) \leqslant \min(b, d)$.

(45) For all elements a, b, c of $\mathbb{R}$ such that $a \leqslant b$ holds $\min(a, c) \leqslant \min(b, c)$.

(46) If $A \subseteq B$, then $A \cap D \subseteq B \cap D$.

(47) If $A \subseteq B$ and $D \subseteq D_1$, then $A \cap D \subseteq B \cap D_1$.

(48) If $A \subseteq B$, then $A \cap B = A$.

(49) If $A \subseteq B$ and $A \subseteq D$ and $B \cap D = E$, then $A = E$.

(50) If $A \cap B \cup A \cap D = A$, then $A \subseteq B \cup D$.

(51) If $A \subseteq B$ and $B \cap D = E$, then $A \cap D = E$.

(52) If $A \subseteq E$, then $A = E$.

(53) $A \cup B = E$ iff $A = E$ and $B = E$.

(54) $A = B \cup D$ iff $B \subseteq A$ and $D \subseteq A$ and for all h_1, D_1 such that $B \subseteq D_1$ and $D \subseteq D_1$ holds $A \subseteq D_1$.

(55) $A = B \cap D$ iff $A \subseteq B$ and $A \subseteq D$ and for all h_1, D_1 such that $D_1 \subseteq B$ and $D_1 \subseteq D$ holds $D_1 \subseteq A$.

(56) If $A \subseteq B \cup D$ and $A \cap D = E$, then $A \subseteq B$.

(57) $A \subseteq B$ iff $B^{\mathrm{c}} \subseteq A^{\mathrm{c}}$.

(58) If $A \subseteq B^{\mathrm{c}}$, then $B \subseteq A^{\mathrm{c}}$.

(59) If $A^{\mathrm{c}} \subseteq B$, then $B^{\mathrm{c}} \subseteq A$.

(60) $(A \cup B)^{\mathrm{c}} \subseteq A^{\mathrm{c}}$ and $(A \cup B)^{\mathrm{c}} \subseteq B^{\mathrm{c}}$.

(61) $A^{\mathrm{c}} \subseteq (A \cap B)^{\mathrm{c}}$ and $B^{\mathrm{c}} \subseteq (A \cap B)^{\mathrm{c}}$.

(62) $\text{1-minus}\,\mathrm{EMF}\,C = \mathrm{UMF}\,C$ and $\text{1-minus}\,\mathrm{UMF}\,C = \mathrm{EMF}\,C$.

(63) $E^{\mathrm{c}} = X$ and $X^{\mathrm{c}} = E$.

4. Exclusive Sum, Absolute Difference

Let C be a non empty set, let h, g be membership functions of C, let A be a FuzzySet of C, h, and let B be a FuzzySet of C, g. The functor $A \dot{-} B$ yields a FuzzySet of C, $\max(\min(h, \text{1-minus}\,g), \min(\text{1-minus}\,h, g))$ and is defined as follows:

(Def. 15) $A \dot{-} B = [\!: C, (\max(\min(h, \text{1-minus}\,g), \min(\text{1-minus}\,h, g)))^{\circ}C :\!]$.

The following propositions are true:

(64) $A \dot{-} B = A \cap B^{\mathrm{c}} \cup A^{\mathrm{c}} \cap B$.

(65) $A \dot{-} B = B \dot{-} A$.

(66) $A \dot{-} E = A$ and $E \dot{-} A = A$.

(67) $A \dot{-} X = A^{\rm c}$ and $X \dot{-} A = A^{\rm c}$.

(68) $A \cap B \cup B \cap D \cup D \cap A = (A \cup B) \cap (B \cup D) \cap (D \cup A)$.

(69) $A \cap B \cup A^{\rm c} \cap B^{\rm c} \subseteq (A \dot{-} B)^{\rm c}$.

(70) $(A \dot{-} B) \cup A \cap B \subseteq A \cup B$.

(71) $A \dot{-} A = A \cap A^{\rm c}$.

Let C be a non empty set and let h, g be membership functions of C. The functor $|h - g|$ yields a membership function of C and is defined as follows:

(Def. 16) For every element c of C holds $|h - g|(c) = |h(c) - g(c)|$.

Let C be a non empty set, let h, g be membership functions of C, let A be a FuzzySet of C, h, and let B be a FuzzySet of C, g. The functor $|A - B|$ yielding a FuzzySet of C, $|h - g|$ is defined by:

(Def. 17) $|A - B| = [\!: C, |h - g|^{\circ}C :\!]$.

References

[1] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[2] Czesław Byliński. Some basic properties of sets. *Formalized Mathematics*, 1(**1**):47–53, 1990.

[3] Jarosław Kotowicz. Partial functions from a domain to the set of real numbers. *Formalized Mathematics*, 1(**4**):703–709, 1990.

[4] Jarosław Kotowicz. Real sequences and basic operations on them. *Formalized Mathematics*, 1(**2**):269–272, 1990.

[5] Konrad Raczkowski and Paweł Sadowski. Topological properties of subsets in real numbers. *Formalized Mathematics*, 1(**4**):777–780, 1990.

[6] Andrzej Trybulec and Czesław Byliński. Some properties of real numbers. *Formalized Mathematics*, 1(**3**):445–449, 1990.

[7] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[8] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

[9] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

Received May 18, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Basic Properties of Fuzzy Set Operation and Membership Function

Takashi Mitsuishi
Shinshu University
Nagano

Katsumi Wasaki
Shinshu University
Nagano

Yasunari Shidama
Shinshu University
Nagano

Summary. This article introduces the fuzzy theory. The definition of the difference set, algebraic product and algebraic sum of fuzzy set is shown. In addition, basic properties of those operations are described. Basic properties of fuzzy set are a little different from those of crisp set.

MML Identifier: `FUZZY_2`.

The articles [3], [1], [2], [4], and [5] provide the terminology and notation for this paper.

1. Basic Properties of Membership Function and Difference Set

For simplicity, we follow the rules: C denotes a non empty set, c denotes an element of C, f, h, g, h_1 denote membership functions of C, A denotes a FuzzySet of C, f, B denotes a FuzzySet of C, g, D denotes a FuzzySet of C, h, D_1 denotes a FuzzySet of C, h_1, X denotes a Universal FuzzySet of C, and E denotes an Empty FuzzySet of C.

We now state four propositions:

(1) For every element x of C and for every membership function h of C holds $0 \leqslant h(x)$ and $h(x) \leqslant 1$.

(2) For every element x of C holds $(\mathrm{EMF}\,C)(x) = 0$ and $(\mathrm{UMF}\,C)(x) = 1$.

(3) For every c such that $f(c) \leqslant h(c)$ holds $(\max(f, \min(g, h)))(c) = (\min(\max(f, g), h))(c)$.

(4) If $A \subseteq D$, then $A \cup B \cap D = (A \cup B) \cap D$.

ISSN 1426–2630

Let C be a non empty set, let f, g be membership functions of C, let A be a FuzzySet of C, f, and let B be a FuzzySet of C, g. The functor $A \setminus B$ yielding a FuzzySet of C, $\min(f, 1\text{-minus}\, g)$ is defined as follows:

(Def. 1) $A \setminus B = [\!: C, (\min(f, 1\text{-minus}\, g))^{\circ}C :\!]$.

Next we state a number of propositions:

(5) $A \setminus B = A \cap B^{\mathrm{c}}$.

(6) $1\text{-minus} \min(f, 1\text{-minus}\, g) = \max(1\text{-minus}\, f, g)$.

(7) $(A \setminus B)^{\mathrm{c}} = A^{\mathrm{c}} \cup B$.

(8) For every c such that $f(c) \leqslant g(c)$ holds $(\min(f, 1\text{-minus}\, h))(c) \leqslant (\min(g, 1\text{-minus}\, h))(c)$.

(9) If $A \subseteq B$, then $A \setminus D \subseteq B \setminus D$.

(10) For every c such that $f(c) \leqslant g(c)$ holds $(\min(h, 1\text{-minus}\, g))(c) \leqslant (\min(h, 1\text{-minus}\, f))(c)$.

(11) If $A \subseteq B$, then $D \setminus B \subseteq D \setminus A$.

(12) For every c such that $f(c) \leqslant g(c)$ and $h(c) \leqslant h_1(c)$ holds $(\min(f, 1\text{-minus}\, h_1))(c) \leqslant (\min(g, 1\text{-minus}\, h))(c)$.

(13) If $A \subseteq B$ and $D \subseteq D_1$, then $A \setminus D_1 \subseteq B \setminus D$.

(14) For every c holds $(\min(f, 1\text{-minus}\, g))(c) \leqslant f(c)$.

(15) $A \setminus B \subseteq A$.

(16) For every c holds $(\min(f, 1\text{-minus}\, g))(c) \leqslant (\max(\min(f, 1\text{-minus}\, g), \min(1\text{-minus}\, f, g)))(c)$.

(17) $A \setminus B \subseteq A \dot{-} B$.

(18) $A \setminus E = A$.

(19) $E \setminus A = E$.

(20) For every c holds $(\min(f, 1\text{-minus}\, g))(c) \leqslant (\min(f, 1\text{-minus} \min(f, g)))(c)$.

(21) $A \setminus B \subseteq A \setminus A \cap B$.

(22) For every c holds $(\max(\min(f, g), \min(f, 1\text{-minus}\, g)))(c) \leqslant f(c)$.

(23) For every c holds $(\max(f, \min(g, 1\text{-minus}\, f)))(c) \leqslant (\max(f, g))(c)$.

(24) $A \cup (B \setminus A) \subseteq A \cup B$.

(25) $A \cap B \cup (A \setminus B) \subseteq A$.

(26) $\min(f, 1\text{-minus} \min(g, 1\text{-minus}\, h)) = \max(\min(f, 1\text{-minus}\, g), \min(f, h))$.

(27) $A \setminus (B \setminus D) = (A \setminus B) \cup A \cap D$.

(28) For every c holds $(\min(f, g))(c) \leqslant (\min(f, 1\text{-minus} \min(f, 1\text{-minus}\, g)))(c)$.

(29) $A \cap B \subseteq A \setminus (A \setminus B)$.

(30) For every c holds $(\min(f, 1\text{-minus}\, g))(c) \leqslant (\min(\max(f, g), 1\text{-minus}\, g))(c)$.

(31) $A \setminus B \subseteq (A \cup B) \setminus B$.

(32) $\min(f, 1\text{-minus} \max(g, h)) = \min(\min(f, 1\text{-minus}\, g), \min(f, 1\text{-minus}\, h))$.

(33) $A \setminus (B \cup D) = (A \setminus B) \cap (A \setminus D)$.

(34) $\min(f, \text{1-minus} \min(g, h)) = \max(\min(f, \text{1-minus}\, g), \min(f, \text{1-minus}\, h))$.

(35) $A \setminus B \cap D = (A \setminus B) \cup (A \setminus D)$.

(36) $\min(\min(f, \text{1-minus}\, g), \text{1-minus}\, h) = \min(f, \text{1-minus} \max(g, h))$.

(37) $A \setminus B \setminus D = A \setminus (B \cup D)$.

(38) For every c holds $(\min(\max(f, g), \text{1-minus} \min(f, g)))(c) \geqslant (\max(\min(f, \text{1-minus}\, g), \min(g, \text{1-minus}\, f)))(c)$.

(39) $(A \setminus B) \cup (B \setminus A) \subseteq (A \cup B) \setminus A \cap B$.

(40) $\min(\max(f, g), \text{1-minus}\, h) = \max(\min(f, \text{1-minus}\, h), \min(g, \text{1-minus}\, h))$.

(41) $(A \cup B) \setminus D = (A \setminus D) \cup (B \setminus D)$.

(42) For every c such that $(\min(f, \text{1-minus}\, g))(c) \leqslant h(c)$ and $(\min(g, \text{1-minus}\, f))(c) \leqslant h(c)$ holds $(\max(\min(f, \text{1-minus}\, g), \min(\text{1-minus}\, f, g)))(c) \leqslant h(c)$.

(43) If $A \setminus B \subseteq D$ and $B \setminus A \subseteq D$, then $A \dot{-} B \subseteq D$.

(44) $A \cap (B \setminus D) = A \cap B \setminus D$.

(45) For every c holds $(\min(f, \min(g, \text{1-minus}\, h)))(c) \leqslant (\min(\min(f, g), \text{1-minus} \min(f, h)))(c)$.

(46) $A \cap (B \setminus D) \subseteq A \cap B \setminus A \cap D$.

(47) For every c holds $(\min(\max(f, g), \text{1-minus} \min(f, g)))(c) \geqslant (\max(\min(f, \text{1-minus}\, g), \min(\text{1-minus}\, f, g)))(c)$.

(48) $A \dot{-} B \subseteq (A \cup B) \setminus A \cap B$.

(49) For every c holds $(\max(\min(f, g), \text{1-minus} \max(f, g)))(c) \leqslant (\text{1-minus} \max(\min(f, \text{1-minus}\, g), \min(\text{1-minus}\, f, g)))(c)$.

(50) $A \cap B \cup (A \cup B)^{\mathrm{c}} \subseteq (A \dot{-} B)^{\mathrm{c}}$.

(51) $\min(\max(\min(f, \text{1-minus}\, g), \min(\text{1-minus}\, f, g)), \text{1-minus}\, h) = \max(\min(f, \text{1-minus} \max(g, h)), \min(g, \text{1-minus} \max(f, h)))$.

(52) $(A \dot{-} B) \setminus D = (A \setminus (B \cup D)) \cup (B \setminus (A \cup D))$.

(53) For every c holds $(\min(f, \text{1-minus} \max(\min(g, \text{1-minus}\, h), \min(\text{1-minus}\, g, h))))(c) \geqslant (\max(\min(f, \text{1-minus} \max(g, h)), \min(\min(f, g), h)))(c)$.

(54) $(A \setminus (B \cup D)) \cup A \cap B \cap D \subseteq A \setminus (B \dot{-} D)$.

(55) For every c such that $f(c) \leqslant g(c)$ holds $g(c) \geqslant (\max(f, \min(g, \text{1-minus}\, f)))(c)$.

(56) If $A \subseteq B$, then $A \cup (B \setminus A) \subseteq B$.

(57) For every c holds $(\max(f, g))(c) \geqslant (\max(\max(\min(f, \text{1-minus}\, g), \min(\text{1-minus}\, f, g)), \min(f, g)))(c)$.

(58) $(A \dot{-} B) \cup A \cap B \subseteq A \cup B$.

(59) If $\min(f, \text{1-minus}\, g) = \mathrm{EMF}\, C$, then for every c holds $f(c) \leqslant g(c)$.

(60) If $A \setminus B = E$, then $A \subseteq B$.

(61) If $\min(f, g) = \mathrm{EMF}\, C$, then $\min(f, \text{1-minus}\, g) = f$.

(62) If $A \cap B = E$, then $A \setminus B = A$.

2. Algebraic Product and Algebraic Sum

Let C be a non empty set and let h, g be membership functions of C. The functor $h \cdot g$ yielding a membership function of C is defined as follows:

(Def. 2) For every element c of C holds $(h \cdot g)(c) = h(c) \cdot g(c)$.

Let C be a non empty set and let h, g be membership functions of C. The functor $h \oplus g$ yielding a membership function of C is defined as follows:

(Def. 3) For every element c of C holds $(h \oplus g)(c) = (h(c) + g(c)) - h(c) \cdot g(c)$.

Let C be a non empty set, let h, g be membership functions of C, let A be a FuzzySet of C, h, and let B be a FuzzySet of C, g. The functor $A \cdot B$ yields a FuzzySet of C, $h \cdot g$ and is defined as follows:

(Def. 4) $A \cdot B = [\!: C, (h \cdot g)^{\circ} C :\!]$.

Let C be a non empty set, let h, g be membership functions of C, let A be a FuzzySet of C, h, and let B be a FuzzySet of C, g. The functor $A \oplus B$ yielding a FuzzySet of C, $h \oplus g$ is defined by:

(Def. 5) $A \oplus B = [\!: C, (h \oplus g)^{\circ} C :\!]$.

We now state a number of propositions:

(63) For every c holds $(f \cdot f)(c) \leqslant f(c)$ and $(f \oplus f)(c) \geqslant f(c)$.

(64) $A \cdot A \subseteq A$ and $A \subseteq A \oplus A$.

(65) $f \cdot g = g \cdot f$ and $f \oplus g = g \oplus f$.

(66) $A \cdot B = B \cdot A$ and $A \oplus B = B \oplus A$.

(67) $(f \cdot g) \cdot h = f \cdot (g \cdot h)$.

(68) $(A \cdot B) \cdot D = A \cdot (B \cdot D)$.

(69) $(f \oplus g) \oplus h = f \oplus (g \oplus h)$.

(70) $(A \oplus B) \oplus D = A \oplus (B \oplus D)$.

(71) For every c holds $(f \cdot (f \oplus g))(c) \leqslant f(c)$ and $(f \oplus f \cdot g)(c) \geqslant f(c)$.

(72) $A \cdot (A \oplus B) \subseteq A$ and $A \subseteq A \oplus A \cdot B$.

(73) For every c holds $(f \cdot (g \oplus h))(c) \leqslant (f \cdot g \oplus f \cdot h)(c)$.

(74) $A \cdot (B \oplus D) \subseteq A \cdot B \oplus A \cdot D$.

(75) For every c holds $((f \oplus g) \cdot (f \oplus h))(c) \leqslant (f \oplus g \cdot h)(c)$.

(76) $(A \oplus B) \cdot (A \oplus D) \subseteq A \oplus B \cdot D$.

(77) 1-minus $f \cdot g =$ 1-minus $f \oplus$ 1-minus g.

(78) $(A \cdot B)^{\rm c} = A^{\rm c} \oplus B^{\rm c}$.

(79) 1-minus $f \oplus g =$ 1-minus $f \cdot$ 1-minus g.

(80) $(A \oplus B)^{\rm c} = A^{\rm c} \cdot B^{\rm c}$.

(81) $f \oplus g = \text{1-minus}\,\text{1-minus}\,f \cdot \text{1-minus}\,g$.

(82) $A \oplus B = (A^{\rm c} \cdot B^{\rm c})^{\rm c}$.

(83) $f \cdot \mathrm{EMF}\,C = \mathrm{EMF}\,C$ and $f \cdot \mathrm{UMF}\,C = f$.

(84) $A \cdot E = E$ and $A \cdot X = A$.

(85) $f \oplus \mathrm{EMF}\,C = f$ and $f \oplus \mathrm{UMF}\,C = \mathrm{UMF}\,C$.

(86) $A \oplus E = A$ and $A \oplus X = X$.

(87) For every c holds $(\mathrm{EMF}\,C)(c) \leqslant (f \cdot \text{1-minus}\,f)(c)$.

(88) For every c holds $(\mathrm{UMF}\,C)(c) \geqslant (f \oplus \text{1-minus}\,f)(c)$.

(89) $E \subseteq A \cdot A^{\rm c}$ and $A \oplus A^{\rm c} \subseteq X$.

(90) For every c holds $(f \cdot g)(c) \leqslant (\min(f, g))(c)$.

(91) $A \cdot B \subseteq A \cap B$.

(92) For every c holds $(\max(f, g))(c) \leqslant (f \oplus g)(c)$.

(93) $A \cup B \subseteq A \oplus B$.

(94) For all real numbers a, b, c such that $0 \leqslant c$ holds $c \cdot \max(a, b) = \max(c \cdot a, c \cdot b)$ and $c \cdot \min(a, b) = \min(c \cdot a, c \cdot b)$.

(95) For all real numbers a, b, c holds $c + \max(a, b) = \max(c + a, c + b)$ and $c + \min(a, b) = \min(c + a, c + b)$.

(96) $f \cdot \max(g, h) = \max(f \cdot g, f \cdot h)$.

(97) $f \cdot \min(g, h) = \min(f \cdot g, f \cdot h)$.

(98) $A \cdot (B \cap D) = (A \cdot B) \cap (A \cdot D)$ and $A \cdot (B \cup D) = A \cdot B \cup A \cdot D$.

(99) $f \oplus \max(g, h) = \max(f \oplus g, f \oplus h)$.

(100) $f \oplus \min(g, h) = \min(f \oplus g, f \oplus h)$.

(101) $A \oplus (B \cup D) = (A \oplus B) \cup (A \oplus D)$ and $A \oplus B \cap D = (A \oplus B) \cap (A \oplus D)$.

(102) For every c holds $(\max(f, g) \cdot \max(f, h))(c) \leqslant (\max(f, g \cdot h))(c)$.

(103) For every c holds $(\min(f, g) \cdot \min(f, h))(c) \leqslant (\min(f, g \cdot h))(c)$.

(104) $(A \cup B) \cdot (A \cup D) \subseteq A \cup B \cdot D$ and $(A \cap B) \cdot (A \cap D) \subseteq A \cap (B \cdot D)$.

(105) For every element c of C and for all membership functions f, g of C holds $(f \oplus g)(c) = 1 - (1 - f(c)) \cdot (1 - g(c))$.

(106) For every c holds $(\max(f, g \oplus h))(c) \leqslant (\max(f, g) \oplus \max(f, h))(c)$.

(107) For every c holds $(\min(f, g \oplus h))(c) \leqslant (\min(f, g) \oplus \min(f, h))(c)$.

(108) $A \cup (B \oplus D) \subseteq (A \cup B) \oplus (A \cup D)$ and $A \cap (B \oplus D) \subseteq A \cap B \oplus A \cap D$.

References

[1] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[2] Jarosław Kotowicz. Real sequences and basic operations on them. *Formalized Mathematics*, 1(**2**):269–272, 1990.

[3] Takashi Mitsuishi, Noboru Endou, and Yasunari Shidama. The concept of fuzzy set and membership function and basic properties of fuzzy set operation. *Formalized Mathematics*, 9(**2**):351–356, 2001.

[4] Andrzej Trybulec and Czesław Byliński. Some properties of real numbers. *Formalized Mathematics*, 1(**3**):445–449, 1990.

[5] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

Received May 22, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

The Hahn Banach Theorem in the Vector Space over the Field of Complex Numbers

Anna Justyna Milewska
University of Białystok

Summary. This article contains the Hahn Banach theorem in the vector space over the field of complex numbers.

MML Identifier: HAHNBAN1.

The articles [8], [7], [1], [5], [2], [6], [9], [3], [14], [10], [12], [13], [4], and [11] provide the terminology and notation for this paper.

1. Preliminaries

The following propositions are true:

(1) For every element z of $\mathbb{C}$ holds $||z|| = |z|$.

(2) For all elements x_1, y_1, x_2, y_2 of $\mathbb{R}$ holds $(x_1 + y_1 i) \cdot (x_2 + y_2 i) = (x_1 \cdot x_2 - y_1 \cdot y_2) + (x_1 \cdot y_2 + x_2 \cdot y_1)i$.

(3) For every real number r holds $(r + 0i) \cdot i = 0 + ri$.

(4) For every real number r holds $|r + 0i| = |r|$.

(5) For every element z of $\mathbb{C}$ such that $|z| \neq 0$ holds $|z| + 0i = \frac{z^*}{|z|+0i} \cdot z$.

2. Some Facts on the Field of Complex Numbers

Let x, y be real numbers. The functor $x + yi_{\mathbb{C}_{\mathrm{F}}}$ yielding an element of $\mathbb{C}_{\mathrm{F}}$ is defined by:

(Def. 1) $x + yi_{\mathbb{C}_{\mathrm{F}}} = x + yi$.

ISSN 1426–2630

The element $i_{\mathbb{C}_{\rm F}}$ of $\mathbb{C}_{\rm F}$ is defined by:

(Def. 2) $i_{\mathbb{C}_{\rm F}} = i$.

One can prove the following propositions:

(6) $i_{\mathbb{C}_{\rm F}} = 0 + 1i$ and $i_{\mathbb{C}_{\rm F}} = 0 + 1i_{\mathbb{C}_{\rm F}}$.

(7) $|i_{\mathbb{C}_{\rm F}}| = 1$.

(8) $i_{\mathbb{C}_{\rm F}} \cdot i_{\mathbb{C}_{\rm F}} = -\mathbf{1}_{\mathbb{C}_{\rm F}}$.

(9) $(-\mathbf{1}_{\mathbb{C}_{\rm F}}) \cdot -\mathbf{1}_{\mathbb{C}_{\rm F}} = \mathbf{1}_{\mathbb{C}_{\rm F}}$.

(10) For all real numbers x_1, y_1, x_2, y_2 holds $(x_1 + y_1 i_{\mathbb{C}_{\rm F}}) + (x_2 + y_2 i_{\mathbb{C}_{\rm F}}) = (x_1 + x_2) + (y_1 + y_2) i_{\mathbb{C}_{\rm F}}$.

(11) For all real numbers x_1, y_1, x_2, y_2 holds $(x_1 + y_1 i_{\mathbb{C}_{\rm F}}) \cdot (x_2 + y_2 i_{\mathbb{C}_{\rm F}}) = (x_1 \cdot x_2 - y_1 \cdot y_2) + (x_1 \cdot y_2 + x_2 \cdot y_1) i_{\mathbb{C}_{\rm F}}$.

(12) For every element z of the carrier of $\mathbb{C}_{\rm F}$ holds $||z|| = |z|$.

(13) For every real number r holds $|r + 0i_{\mathbb{C}_{\rm F}}| = |r|$.

(14) For every real number r holds $(r + 0i_{\mathbb{C}_{\rm F}}) \cdot i_{\mathbb{C}_{\rm F}} = 0 + r i_{\mathbb{C}_{\rm F}}$.

Let z be an element of the carrier of $\mathbb{C}_{\rm F}$. The functor $\Re(z)$ yields a real number and is defined as follows:

(Def. 3) There exists an element z' of $\mathbb{C}$ such that $z = z'$ and $\Re(z) = \Re(z')$.

Let z be an element of the carrier of $\mathbb{C}_{\rm F}$. The functor $\Im(z)$ yields a real number and is defined as follows:

(Def. 4) There exists an element z' of $\mathbb{C}$ such that $z = z'$ and $\Im(z) = \Im(z')$.

The following propositions are true:

(15) For all real numbers x, y holds $\Re(x + y i_{\mathbb{C}_{\rm F}}) = x$ and $\Im(x + y i_{\mathbb{C}_{\rm F}}) = y$.

(16) For all elements x, y of the carrier of $\mathbb{C}_{\rm F}$ holds $\Re(x + y) = \Re(x) + \Re(y)$ and $\Im(x + y) = \Im(x) + \Im(y)$.

(17) For all elements x, y of the carrier of $\mathbb{C}_{\rm F}$ holds $\Re(x \cdot y) = \Re(x) \cdot \Re(y) - \Im(x) \cdot \Im(y)$ and $\Im(x \cdot y) = \Re(x) \cdot \Im(y) + \Re(y) \cdot \Im(x)$.

(18) For every element z of the carrier of $\mathbb{C}_{\rm F}$ holds $\Re(z) \leqslant |z|$.

(19) For every element z of the carrier of $\mathbb{C}_{\rm F}$ holds $\Im(z) \leqslant |z|$.

3. Functionals of Vector Space

Let K be a 1-sorted structure and let V be a vector space structure over K.

(Def. 5) A function from the carrier of V into the carrier of K is said to be a functional in V.

Let K be a non empty loop structure, let V be a non empty vector space structure over K, and let f, g be functionals in V. The functor $f + g$ yielding a functional in V is defined by:

(Def. 6) For every element x of the carrier of V holds $(f+g)(x) = f(x) + g(x)$.

Let K be a non empty loop structure, let V be a non empty vector space structure over K, and let f be a functional in V. The functor $-f$ yielding a functional in V is defined by:

(Def. 7) For every element x of the carrier of V holds $(-f)(x) = -f(x)$.

Let K be a non empty loop structure, let V be a non empty vector space structure over K, and let f, g be functionals in V. The functor $f - g$ yielding a functional in V is defined by:

(Def. 8) $f - g = f + -g$.

Let K be a non empty groupoid, let V be a non empty vector space structure over K, let v be an element of the carrier of K, and let f be a functional in V. The functor $v \cdot f$ yields a functional in V and is defined by:

(Def. 9) For every element x of the carrier of V holds $(v \cdot f)(x) = v \cdot f(x)$.

Let K be a non empty zero structure and let V be a vector space structure over K. The functor 0Functional V yields a functional in V and is defined as follows:

(Def. 10) 0Functional $V = \Omega_V \longmapsto 0_K$.

Let K be a non empty loop structure, let V be a non empty vector space structure over K, and let F be a functional in V. We say that F is additive if and only if:

(Def. 11) For all vectors x, y of V holds $F(x+y) = F(x) + F(y)$.

Let K be a non empty groupoid, let V be a non empty vector space structure over K, and let F be a functional in V. We say that F is homogeneous if and only if:

(Def. 12) For every vector x of V and for every scalar r of V holds $F(r \cdot x) = r \cdot F(x)$.

Let K be a non empty zero structure, let V be a non empty vector space structure over K, and let F be a functional in V. We say that F is 0-preserving if and only if:

(Def. 13) $F(0_V) = 0_K$.

Let K be an add-associative right zeroed right complementable Abelian associative left unital distributive non empty double loop structure and let V be a vector space over K. Note that every functional in V which is homogeneous is also 0-preserving.

Let K be a right zeroed non empty loop structure and let V be a non empty vector space structure over K. Note that 0Functional V is additive.

Let K be an add-associative right zeroed right complementable right distributive non empty double loop structure and let V be a non empty vector space structure over K. Observe that 0Functional V is homogeneous.

Let K be a non empty zero structure and let V be a non empty vector space structure over K. Observe that 0Functional V is 0-preserving.

Let K be an add-associative right zeroed right complementable right distributive non empty double loop structure and let V be a non empty vector space structure over K. Observe that there exists a functional in V which is additive, homogeneous, and 0-preserving.

The following propositions are true:

(20) Let K be an Abelian non empty loop structure, V be a non empty vector space structure over K, and f, g be functionals in V. Then $f+g=g+f$.

(21) Let K be an add-associative non empty loop structure, V be a non empty vector space structure over K, and f, g, h be functionals in V. Then $(f+g)+h=f+(g+h)$.

(22) Let K be a non empty zero structure, V be a non empty vector space structure over K, and x be an element of the carrier of V. Then $(0\mathrm{Functional}\,V)(x)=0_K$.

(23) Let K be a right zeroed non empty loop structure, V be a non empty vector space structure over K, and f be a functional in V. Then $f+0\mathrm{Functional}\,V=f$.

(24) Let K be an add-associative right zeroed right complementable non empty loop structure, V be a non empty vector space structure over K, and f be a functional in V. Then $f-f=0\mathrm{Functional}\,V$.

(25) Let K be a right distributive non empty double loop structure, V be a non empty vector space structure over K, r be an element of the carrier of K, and f, g be functionals in V. Then $r\cdot(f+g)=r\cdot f+r\cdot g$.

(26) Let K be a left distributive non empty double loop structure, V be a non empty vector space structure over K, r, s be elements of the carrier of K, and f be a functional in V. Then $(r+s)\cdot f=r\cdot f+s\cdot f$.

(27) Let K be an associative non empty groupoid, V be a non empty vector space structure over K, r, s be elements of the carrier of K, and f be a functional in V. Then $(r\cdot s)\cdot f=r\cdot(s\cdot f)$.

(28) Let K be a left unital non empty double loop structure, V be a non empty vector space structure over K, and f be a functional in V. Then $\mathbf{1}_K\cdot f=f$.

Let K be an Abelian add-associative right zeroed right complementable right distributive non empty double loop structure, let V be a non empty vector space structure over K, and let f, g be additive functionals in V. Observe that $f+g$ is additive.

Let K be an Abelian add-associative right zeroed right complementable right distributive non empty double loop structure, let V be a non empty vector space structure over K, and let f be an additive functional in V. One can verify that $-f$ is additive.

Let K be an add-associative right zeroed right complementable right distributive non empty double loop structure, let V be a non empty vector space structure over K, let v be an element of the carrier of K, and let f be an additive functional in V. Observe that $v \cdot f$ is additive.

Let K be an add-associative right zeroed right complementable right distributive non empty double loop structure, let V be a non empty vector space structure over K, and let f, g be homogeneous functionals in V. Observe that $f + g$ is homogeneous.

Let K be an Abelian add-associative right zeroed right complementable right distributive non empty double loop structure, let V be a non empty vector space structure over K, and let f be a homogeneous functional in V. One can check that $-f$ is homogeneous.

Let K be an add-associative right zeroed right complementable right distributive associative commutative non empty double loop structure, let V be a non empty vector space structure over K, let v be an element of the carrier of K, and let f be a homogeneous functional in V. Observe that $v \cdot f$ is homogeneous.

Let K be an add-associative right zeroed right complementable right distributive non empty double loop structure and let V be a non empty vector space structure over K. A linear functional in V is an additive homogeneous functional in V.

4. The Vector Space of Linear Functionals

Let K be an Abelian add-associative right zeroed right complementable right distributive associative commutative non empty double loop structure and let V be a non empty vector space structure over K. The functor V^* yielding a non empty strict vector space structure over K is defined by the conditions (Def. 14).

(Def. 14)(i) For every set x holds $x \in$ the carrier of V^* iff x is a linear functional in V,

(ii) for all linear functionals f, g in V holds (the addition of V^*)$(f, g) = f + g$,

(iii) for every linear functional f in V holds (the reverse-map of V^*)$(f) = -f$,

(iv) the zero of $V^* =$ 0Functional V, and

(v) for every linear functional f in V and for every element x of the carrier of K holds (the left multiplication of V^*)$(x, f) = x \cdot f$.

Let K be an Abelian add-associative right zeroed right complementable right distributive associative commutative non empty double loop structure and let V be a non empty vector space structure over K. One can check that V^* is Abelian.

Let K be an Abelian add-associative right zeroed right complementable right distributive associative commutative non empty double loop structure and let

V be a non empty vector space structure over K. One can verify the following observations:

- $*$ V^* is add-associative,
- $*$ V^* is right zeroed, and
- $*$ V^* is right complemented.

Let K be an Abelian add-associative right zeroed right complementable left unital distributive associative commutative non empty double loop structure and let V be a non empty vector space structure over K. One can check that V^* is vector space-like.

5. Semi Norm of Vector Space

Let K be a 1-sorted structure and let V be a vector space structure over K.

(Def. 15) A function from the carrier of V into $\mathbb{R}$ is said to be a RFunctional of V.

Let K be a 1-sorted structure, let V be a non empty vector space structure over K, and let F be a RFunctional of V. We say that F is subadditive if and only if:

(Def. 16) For all vectors x, y of V holds $F(x+y) \leqslant F(x)+F(y)$.

Let K be a 1-sorted structure, let V be a non empty vector space structure over K, and let F be a RFunctional of V. We say that F is additive if and only if:

(Def. 17) For all vectors x, y of V holds $F(x+y) = F(x)+F(y)$.

Let V be a non empty vector space structure over $\mathbb{C}_{\rm F}$ and let F be a RFunctional of V. We say that F is Real-homogeneous if and only if:

(Def. 18) For every vector v of V and for every real number r holds $F((r+0i_{\mathbb{C}_{\rm F}}) \cdot v) = r \cdot F(v)$.

One can prove the following proposition

(29) Let V be a vector space-like non empty vector space structure over $\mathbb{C}_{\rm F}$ and F be a RFunctional of V. Suppose F is Real-homogeneous. Let v be a vector of V and r be a real number. Then $F((0+ri_{\mathbb{C}_{\rm F}}) \cdot v) = r \cdot F(i_{\mathbb{C}_{\rm F}} \cdot v)$.

Let V be a non empty vector space structure over $\mathbb{C}_{\rm F}$ and let F be a RFunctional of V. We say that F is homogeneous if and only if:

(Def. 19) For every vector v of V and for every scalar r of V holds $F(r \cdot v) = |r| \cdot F(v)$.

Let K be a 1-sorted structure, let V be a vector space structure over K, and let F be a RFunctional of V. We say that F is 0-preserving if and only if:

(Def. 20) $F(0_V) = 0$.

Let K be a 1-sorted structure and let V be a non empty vector space structure over K. One can verify that every RFunctional of V which is additive is also subadditive.

Let V be a vector space over $\mathbb{C}_{\rm F}$. Note that every RFunctional of V which is Real-homogeneous is also 0-preserving.

Let K be a 1-sorted structure and let V be a vector space structure over K. The functor 0RFunctional V yielding a RFunctional of V is defined as follows:

(Def. 21) $\quad$ 0RFunctional $V = \Omega_V \longmapsto 0$.

Let K be a 1-sorted structure and let V be a non empty vector space structure over K. Note that 0RFunctional V is additive and 0RFunctional V is 0-preserving.

Let V be a non empty vector space structure over $\mathbb{C}_{\rm F}$. Note that 0RFunctional V is Real-homogeneous and 0RFunctional V is homogeneous.

Let K be a 1-sorted structure and let V be a non empty vector space structure over K. Note that there exists a RFunctional of V which is additive and 0-preserving.

Let V be a non empty vector space structure over $\mathbb{C}_{\rm F}$. One can check that there exists a RFunctional of V which is additive, Real-homogeneous, and homogeneous.

Let V be a non empty vector space structure over $\mathbb{C}_{\rm F}$. A Semi-Norm of V is a subadditive homogeneous RFunctional of V.

6. The Hahn Banach Theorem

Let V be a non empty vector space structure over $\mathbb{C}_{\rm F}$. The functor RealVS V yielding a strict RLS structure is defined by the conditions (Def. 22).

(Def. 22)(i) $\quad$ The loop structure of RealVS V = the loop structure of V, and

(ii) $\quad$ for every real number r and for every vector v of V holds (the external multiplication of RealVS $V)(r, v) = (r + 0i_{\mathbb{C}_{\rm F}}) \cdot v$.

Let V be a non empty vector space structure over $\mathbb{C}_{\rm F}$. Observe that RealVS V is non empty.

Let V be an Abelian non empty vector space structure over $\mathbb{C}_{\rm F}$. Observe that RealVS V is Abelian.

Let V be an add-associative non empty vector space structure over $\mathbb{C}_{\rm F}$. One can check that RealVS V is add-associative.

Let V be a right zeroed non empty vector space structure over $\mathbb{C}_{\rm F}$. Note that RealVS V is right zeroed.

Let V be a right complementable non empty vector space structure over $\mathbb{C}_{\rm F}$. One can check that RealVS V is right complementable.

Let V be a vector space-like non empty vector space structure over $\mathbb{C}_{\rm F}$. Note that RealVS V is real linear space-like.

One can prove the following three propositions:

(30) For every non empty vector space V over $\mathbb{C}_{\rm F}$ and for every subspace M of V holds RealVS M is a subspace of RealVS V.

(31) For every non empty vector space structure V over $\mathbb{C}_{\rm F}$ holds every RFunctional of V is a functional in RealVS V.

(32) For every non empty vector space V over $\mathbb{C}_{\rm F}$ holds every Semi-Norm of V is a Banach functional in RealVS V.

Let V be a non empty vector space structure over $\mathbb{C}_{\rm F}$ and let l be a functional in V. The functor projRe l yielding a functional in RealVS V is defined by:

(Def. 23) For every element i of the carrier of V holds $(\mathrm{projRe}\, l)(i) = \Re(l(i))$.

Let V be a non empty vector space structure over $\mathbb{C}_{\rm F}$ and let l be a functional in V. The functor projIm l yields a functional in RealVS V and is defined as follows:

(Def. 24) For every element i of the carrier of V holds $(\mathrm{projIm}\, l)(i) = \Im(l(i))$.

Let V be a non empty vector space structure over $\mathbb{C}_{\rm F}$ and let l be a functional in RealVS V. The functor $l_{\mathbb{R}\to\mathbb{C}}$ yielding a RFunctional of V is defined by:

(Def. 25) $l_{\mathbb{R}\to\mathbb{C}} = l$.

Let V be a non empty vector space structure over $\mathbb{C}_{\rm F}$ and let l be a RFunctional of V. The functor $l_{\mathbb{C}\to\mathbb{R}}$ yields a functional in RealVS V and is defined by:

(Def. 26) $l_{\mathbb{C}\to\mathbb{R}} = l$.

Let V be a non empty vector space over $\mathbb{C}_{\rm F}$ and let l be an additive functional in RealVS V. One can check that $l_{\mathbb{R}\to\mathbb{C}}$ is additive.

Let V be a non empty vector space over $\mathbb{C}_{\rm F}$ and let l be an additive RFunctional of V. Observe that $l_{\mathbb{C}\to\mathbb{R}}$ is additive.

Let V be a non empty vector space over $\mathbb{C}_{\rm F}$ and let l be a homogeneous functional in RealVS V. Observe that $l_{\mathbb{R}\to\mathbb{C}}$ is Real-homogeneous.

Let V be a non empty vector space over $\mathbb{C}_{\rm F}$ and let l be a Real-homogeneous RFunctional of V. One can verify that $l_{\mathbb{C}\to\mathbb{R}}$ is homogeneous.

Let V be a non empty vector space structure over $\mathbb{C}_{\rm F}$ and let l be a RFunctional of V. The functor i-shift l yields a RFunctional of V and is defined by:

(Def. 27) For every element v of the carrier of V holds $(\text{i-shift}\, l)(v) = l(i_{\mathbb{C}_{\rm F}} \cdot v)$.

Let V be a non empty vector space structure over $\mathbb{C}_{\rm F}$ and let l be a functional in RealVS V. The functor prodReIm l yielding a functional in V is defined as follows:

(Def. 28) For every element v of the carrier of V holds $(\mathrm{prodReIm}\, l)(v) = (l_{\mathbb{R}\to\mathbb{C}})(v) + (-(\text{i-shift}\, l_{\mathbb{R}\to\mathbb{C}})(v))i_{\mathbb{C}_{\rm F}}$.

The following four propositions are true:

(33) Let V be a non empty vector space over $\mathbb{C}_{\rm F}$ and l be a linear functional in V. Then projRe l is a linear functional in RealVS V.

(34) Let V be a non empty vector space over $\mathbb{C}_{\rm F}$ and l be a linear functional in V. Then projIm l is a linear functional in RealVS V.

(35) Let V be a non empty vector space over $\mathbb{C}_{\rm F}$ and l be a linear functional in RealVS V. Then prodReIm l is a linear functional in V.

(36) Let V be a non empty vector space over $\mathbb{C}_{\rm F}$, p be a Semi-Norm of V, M be a subspace of V, and l be a linear functional in M. Suppose that for every vector e of M and for every vector v of V such that $v = e$ holds $|l(e)| \leqslant p(v)$. Then there exists a linear functional L in V such that $L{\restriction}$the carrier of $M = l$ and for every vector e of V holds $|L(e)| \leqslant p(e)$.

References

[1] Czesław Byliński. The complex numbers. *Formalized Mathematics*, 1(**3**):507–513, 1990.

[2] Czesław Byliński. Functions from a set to a set. *Formalized Mathematics*, 1(**1**):153–164, 1990.

[3] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[4] Eugeniusz Kusak, Wojciech Leończuk, and Michał Muzalewski. Abelian groups, fields and vector spaces. *Formalized Mathematics*, 1(**2**):335–342, 1990.

[5] Anna Justyna Milewska. The field of complex numbers. *Formalized Mathematics*, 9(**2**):265–269, 2001.

[6] Bogdan Nowak and Andrzej Trybulec. Hahn-Banach theorem. *Formalized Mathematics*, 4(**1**):29–34, 1993.

[7] Jan Popiołek. Some properties of functions modul and signum. *Formalized Mathematics*, 1(**2**):263–264, 1990.

[8] Wojciech Skaba and Michał Muzalewski. From double loops to fields. *Formalized Mathematics*, 2(**1**):185–191, 1991.

[9] Andrzej Trybulec. Natural transformations. Discrete categories. *Formalized Mathematics*, 2(**4**):467–474, 1991.

[10] Wojciech A. Trybulec. Subspaces and cosets of subspaces in real linear space. *Formalized Mathematics*, 1(**2**):297–301, 1990.

[11] Wojciech A. Trybulec. Subspaces and cosets of subspaces in vector space. *Formalized Mathematics*, 1(**5**):865–870, 1990.

[12] Wojciech A. Trybulec. Vectors in real linear space. *Formalized Mathematics*, 1(**2**):291–296, 1990.

[13] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[14] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

Received May 23, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

The Tichonov Theorem

Bartłomiej Skorulski
University of Białystok

MML Identifier: YELLOW17.

The terminology and notation used here are introduced in the following articles: [15], [11], [1], [5], [7], [4], [3], [13], [8], [10], [16], [14], [12], [6], [9], and [2].

1. Some Properties of Products

One can prove the following propositions:

(1) For every function F and for all sets i, x_1 and for every subset A_1 of $F(i)$ such that $(\mathrm{proj}(F,i))^{-1}(\{x_1\}) \cap (\mathrm{proj}(F,i))^{-1}(A_1) \neq \emptyset$ holds $x_1 \in A_1$.

(2) For all functions F, f and for all sets i, x_1 such that $x_1 \in F(i)$ and $f \in \prod F$ holds $f +\cdot (i, x_1) \in \prod F$.

(3) For every function F and for every set i such that $i \in \mathrm{dom}\, F$ and $\prod F \neq \emptyset$ holds $\mathrm{rng}\,\mathrm{proj}(F,i) = F(i)$.

(4) For every function F and for every set i such that $i \in \mathrm{dom}\, F$ holds $(\mathrm{proj}(F,i))^{-1}(F(i)) = \prod F$.

(5) For all functions F, f and for all sets i, x_1 such that $x_1 \in F(i)$ and $i \in \mathrm{dom}\, F$ and $f \in \prod F$ holds $f +\cdot (i, x_1) \in (\mathrm{proj}(F,i))^{-1}(\{x_1\})$.

(6) Let F, f be functions, i_1, i_2, x_2 be sets, and A_2 be a subset of $F(i_2)$. Suppose $x_2 \in F(i_1)$ and $i_1 \in \mathrm{dom}\, F$ and $f \in \prod F$. If $i_1 \neq i_2$, then $f \in (\mathrm{proj}(F,i_2))^{-1}(A_2)$ iff $f +\cdot (i_1, x_2) \in (\mathrm{proj}(F,i_2))^{-1}(A_2)$.

(7) Let F be a function, i_1, i_2, x_2 be sets, and A_2 be a subset of $F(i_2)$. Suppose $\prod F \neq \emptyset$ and $x_2 \in F(i_1)$ and $i_1 \in \mathrm{dom}\, F$ and $i_2 \in \mathrm{dom}\, F$ and $A_2 \neq F(i_2)$. Then $(\mathrm{proj}(F,i_1))^{-1}(\{x_2\}) \subseteq (\mathrm{proj}(F,i_2))^{-1}(A_2)$ if and only if $i_1 = i_2$ and $x_2 \in A_2$.

The scheme *ElProductEx* deals with a non empty set $\mathcal{A}$, a topological space yielding nonempty many sorted set $\mathcal{B}$ indexed by $\mathcal{A}$, and a binary predicate $\mathcal{P}$, and states that:

> There exists an element f of $\prod \mathcal{B}$ such that for every element i of $\mathcal{A}$ holds $\mathcal{P}[f(i), i]$

provided the parameters have the following property:

- For every element i of $\mathcal{A}$ there exists an element x of $\mathcal{B}(i)$ such that $\mathcal{P}[x, i]$.

One can prove the following propositions:

(8) Let I be a non empty set, J be a topological space yielding nonempty many sorted set indexed by I, i be an element of I, and f be an element of $\prod J$. Then $(\mathrm{proj}(J, i))(f) = f(i)$.

(9) Let I be a non empty set, J be a topological space yielding nonempty many sorted set indexed by I, i be an element of I, x_1 be an element of $J(i)$, and A_1 be a subset of $J(i)$. If $(\mathrm{proj}(J, i))^{-1}(\{x_1\}) \cap (\mathrm{proj}(J, i))^{-1}(A_1) \neq \emptyset$, then $x_1 \in A_1$.

(10) Let I be a non empty set, J be a topological space yielding nonempty many sorted set indexed by I, and i be an element of I. Then $(\mathrm{proj}(J, i))^{-1}(\Omega_{J(i)}) = \Omega_{\prod J}$.

(11) Let I be a non empty set, J be a topological space yielding nonempty many sorted set indexed by I, i be an element of I, x_1 be an element of $J(i)$, and f be an element of $\prod J$. Then $f +\cdot (i, x_1) \in (\mathrm{proj}(J, i))^{-1}(\{x_1\})$.

(12) Let I be a non empty set, J be a topological space yielding nonempty many sorted set indexed by I, i_1, i_2 be elements of I, x_2 be an element of $J(i_1)$, and A_2 be a subset of $J(i_2)$. If $A_2 \neq \Omega_{J(i_2)}$, then $(\mathrm{proj}(J, i_1))^{-1}(\{x_2\}) \subseteq (\mathrm{proj}(J, i_2))^{-1}(A_2)$ iff $i_1 = i_2$ and $x_2 \in A_2$.

(13) Let I be a non empty set, J be a topological space yielding nonempty many sorted set indexed by I, i_1, i_2 be elements of I, x_2 be an element of $J(i_1)$, A_2 be a subset of $J(i_2)$, and f be an element of $\prod J$. If $i_1 \neq i_2$, then $f \in (\mathrm{proj}(J, i_2))^{-1}(A_2)$ iff $f +\cdot (i_1, x_2) \in (\mathrm{proj}(J, i_2))^{-1}(A_2)$.

2. Some Properties of Compact Spaces

One can prove the following three propositions:

(14) Let T be a topological structure and F be a family of subsets of T. Then F is a cover of T if and only if the carrier of $T \subseteq \bigcup F$.

(15) Let T be a non empty topological structure. Then T is compact if and only if for every family F of subsets of T such that F is open and $\Omega_T \subseteq \bigcup F$ there exists a family G of subsets of T such that $G \subseteq F$ and $\Omega_T \subseteq \bigcup G$ and G is finite.

(16) Let T be a non empty topological space and B be a prebasis of T. Then T is compact if and only if for every subset F of B such that $\Omega_T \subseteq \bigcup F$ there exists a finite subset G of F such that $\Omega_T \subseteq \bigcup G$.

3. The Tichonov Theorem

The following propositions are true:

(17) Let I be a non empty set, J be a topological space yielding nonempty many sorted set indexed by I, and A be a set. Suppose $A \in$ the product prebasis for J. Then there exists an element i of I and there exists a subset A_1 of $J(i)$ such that A_1 is open and $(\mathrm{proj}(J,i))^{-1}(A_1) = A$.

(18) Let I be a non empty set, J be a topological space yielding nonempty many sorted set indexed by I, i be an element of I, x_1 be an element of $J(i)$, and A be a set. Suppose $A \in$ the product prebasis for J and $(\mathrm{proj}(J,i))^{-1}(\{x_1\}) \subseteq A$. Then $A = \Omega_{\prod J}$ or there exists a subset A_1 of $J(i)$ such that $A_1 \neq \Omega_{J(i)}$ and $x_1 \in A_1$ and A_1 is open and $A = (\mathrm{proj}(J,i))^{-1}(A_1)$.

(19) Let I be a non empty set, J be a topological space yielding nonempty many sorted set indexed by I, i be an element of I, and F_1 be a non empty family of subsets of $J(i)$. If $\Omega_{J(i)} \subseteq \bigcup F_1$, then $\Omega_{\prod J} \subseteq \bigcup\{(\mathrm{proj}(J,i))^{-1}(A_1) : A_1$ ranges over elements of $F_1\}$.

(20) Let I be a non empty set, J be a topological space yielding nonempty many sorted set indexed by I, i be an element of I, x_1 be an element of $J(i)$, and G be a subset of the product prebasis for J. Suppose $(\mathrm{proj}(J,i))^{-1}(\{x_1\}) \subseteq \bigcup G$ and for every set A such that $A \in$ the product prebasis for J and $A \in G$ holds $(\mathrm{proj}(J,i))^{-1}(\{x_1\}) \not\subseteq A$. Then $\Omega_{\prod J} \subseteq \bigcup G$.

(21) Let I be a non empty set, J be a topological space yielding nonempty many sorted set indexed by I, i be an element of I, and F be a subset of the product prebasis for J. Suppose that for every finite subset G of F holds $\Omega_{\prod J} \not\subseteq \bigcup G$. Let x_1 be an element of $J(i)$ and G be a finite subset of F. Suppose $(\mathrm{proj}(J,i))^{-1}(\{x_1\}) \subseteq \bigcup G$. Then there exists a set A such that $A \in$ the product prebasis for J and $A \in G$ and $(\mathrm{proj}(J,i))^{-1}(\{x_1\}) \subseteq A$.

(22) Let I be a non empty set, J be a topological space yielding nonempty many sorted set indexed by I, i be an element of I, and F be a subset of the product prebasis for J. Suppose that for every finite subset G of F holds $\Omega_{\prod J} \not\subseteq \bigcup G$. Let x_1 be an element of $J(i)$ and G be a finite subset of F. Suppose $(\mathrm{proj}(J,i))^{-1}(\{x_1\}) \subseteq \bigcup G$. Then there exists a subset A_1 of $J(i)$ such that $A_1 \neq \Omega_{J(i)}$ and $x_1 \in A_1$ and $(\mathrm{proj}(J,i))^{-1}(A_1) \in G$ and A_1 is open.

(23) Let I be a non empty set, J be a topological space yielding nonempty many sorted set indexed by I, i be an element of I, and F be a subset of the product prebasis for J. Suppose for every element i of I holds $J(i)$ is compact and for every finite subset G of F holds $\Omega_{\prod J} \not\subseteq \bigcup G$. Then there exists an element x_1 of $J(i)$ such that for every finite subset G of F holds $(\mathrm{proj}(J,i))^{-1}(\{x_1\}) \not\subseteq \bigcup G$.

(24) Let I be a non empty set and J be a topological space yielding nonempty many sorted set indexed by I. If for every element i of I holds $J(i)$ is compact, then $\prod J$ is compact.

References

[1] Grzegorz Bancerek. König's theorem. *Formalized Mathematics*, 1(**3**):589–593, 1990.

[2] Grzegorz Bancerek. The "way-below" relation. *Formalized Mathematics*, 6(**1**):169–176, 1997.

[3] Grzegorz Bancerek and Andrzej Trybulec. Miscellaneous facts about functions. *Formalized Mathematics*, 5(**4**):485–492, 1996.

[4] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[5] Agata Darmochwał. Compact spaces. *Formalized Mathematics*, 1(**2**):383–386, 1990.

[6] Agata Darmochwał. Families of subsets, subspaces and mappings in topological spaces. *Formalized Mathematics*, 1(**2**):257–261, 1990.

[7] Agata Darmochwał. Finite sets. *Formalized Mathematics*, 1(**1**):165–167, 1990.

[8] Mariusz Giero. More on products of many sorted algebras. *Formalized Mathematics*, 5(**4**):621–626, 1996.

[9] Jarosław Gryko. Injective spaces. *Formalized Mathematics*, 7(**1**):57–62, 1998.

[10] Beata Padlewska and Agata Darmochwał. Topological spaces and continuous functions. *Formalized Mathematics*, 1(**1**):223–230, 1990.

[11] Alexander Yu. Shibakov and Andrzej Trybulec. The Cantor set. *Formalized Mathematics*, 5(**2**):233–236, 1996.

[12] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[13] Andrzej Trybulec. Many-sorted sets. *Formalized Mathematics*, 4(**1**):15–22, 1993.

[14] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[15] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

[16] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

Received May 23, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

On the Order-consistent Topology of Complete and Uncomplete Lattices

Ewa Grądzka
University of Białystok

Summary. This paper is a continuation of the formalisation of [5] pp. 108–109. Order-consistent and upper topologies are defined. The theorem that the Scott and the upper topologies are order-consistent is proved. Remark 1.4 and example 1.5(2) are generalized for proving this theorem.

MML Identifier: WAYBEL32.

The terminology and notation used in this paper are introduced in the following papers: [8], [12], [1], [13], [9], [15], [14], [16], [11], [3], [6], [7], [2], [10], and [4].

Let T be a non empty FR-structure. We say that T is upper if and only if:

(Def. 1) $\{-\downarrow x : x$ ranges over elements of $T\}$ is a prebasis of T.

Let us mention that there exists a top-lattice which is Scott, up-complete, and strict.

Let T be a topological space-like non empty reflexive FR-structure. We say that T is order consistent if and only if the condition (Def. 2) is satisfied.

(Def. 2) Let x be an element of T. Then

(i) $\downarrow x = \overline{\{x\}}$, and

(ii) for every eventually-directed net N in T such that $x = \sup N$ and for every neighbourhood V of x holds N is eventually in V.

One can verify that every non empty reflexive topological space-like FR-structure which is trivial is also upper.

Let us mention that there exists a top-lattice which is upper, trivial, up-complete, and strict.

The following propositions are true:

(1) For every upper up-complete non empty top-poset T and for every subset A of T such that A is open holds A is upper.

(2) For every up-complete non empty top-poset T such that T is upper holds T is order consistent.

(3) Let T be a Scott up-complete non empty reflexive transitive antisymmetric FR-structure and x be an element of T. Then $\downarrow x$ is directly closed and lower.

(4) Let T be a Scott up-complete non empty reflexive transitive antisymmetric FR-structure and S be a subset of T. Then S is closed if and only if S is directly closed and lower.

(5) Let T be a Scott up-complete non empty reflexive transitive antisymmetric FR-structure and x be an element of T. Then $\downarrow x$ is closed.

(6) Let S be an up-complete reflexive antisymmetric non empty relational structure and T be a non empty reflexive relational structure. Suppose the relational structure of $S =$ the relational structure of T. Let A be a subset of S and C be a subset of T. If $A = C$ and A is inaccessible, then C is inaccessible.

(7) For every up-complete non empty reflexive transitive antisymmetric relational structure R holds there exists a topological augmentation of R which is Scott.

(8) Let R be an up-complete non empty poset and T be a topological augmentation of R. If T is Scott, then T is correct.

Let R be an up-complete non empty reflexive transitive antisymmetric relational structure. Observe that every topological augmentation of R which is Scott is also correct.

Let R be an up-complete non empty reflexive transitive antisymmetric relational structure. Note that there exists a topological augmentation of R which is Scott and correct.

The following propositions are true:

(9) Let T be a Scott up-complete non empty reflexive transitive antisymmetric FR-structure and x be an element of T. Then $\overline{\{x\}} = \downarrow x$.

(10) Every up-complete Scott non empty top-poset is order consistent.

(11) Let R be an inf-complete semilattice, Z be a net in R, and D be a subset of R. Suppose $D = \{\bigsqcap_R \{Z(k); k$ ranges over elements of the carrier of Z: $k \geqslant j\} : j$ ranges over elements of the carrier of $Z\}$. Then D is non empty and directed.

(12) Let R be an inf-complete semilattice, S be a subset of R, and a be an element of R. If $a \in S$, then $\bigsqcap_R S \leqslant a$.

(13) For every inf-complete semilattice R and for every monotone reflexive net N in R holds $\liminf N = \sup N$.

(14) Let R be an inf-complete semilattice and S be a subset of R. Then $S \in$ the topology of ConvergenceSpace(the Scott convergence of R) if and

only if S is inaccessible and upper.

(15) Let R be an inf-complete up-complete semilattice and T be a topological augmentation of R. If the topology of $T = \sigma(R)$, then T is Scott.

Let R be an inf-complete up-complete semilattice. One can check that there exists a topological augmentation of R which is strict, Scott, and correct.

One can prove the following two propositions:

(16) Let S be an up-complete inf-complete semilattice and T be a Scott topological augmentation of S. Then $\sigma(S) =$ the topology of T.

(17) Every Scott up-complete non empty reflexive transitive antisymmetric FR-structure is a T_0-space.

Let R be an up-complete non empty reflexive transitive antisymmetric relational structure. Note that every topological augmentation of R is up-complete.

The following propositions are true:

(18) Let R be an up-complete non empty reflexive transitive antisymmetric relational structure, T be a Scott topological augmentation of R, x be an element of T, and A be an upper subset of T. If $x \notin A$, then $-{\downarrow}x$ is a neighbourhood of A.

(19) Let R be an up-complete non empty reflexive transitive antisymmetric FR-structure, T be a Scott topological augmentation of R, and S be an upper subset of T. Then there exists a family F of subsets of T such that $S = \bigcap F$ and for every subset X of T such that $X \in F$ holds X is a neighbourhood of S.

(20) Let T be a Scott up-complete non empty reflexive transitive antisymmetric FR-structure and S be a subset of T. Then S is open if and only if S is upper and property(S).

(21) Let R be an up-complete non empty reflexive transitive antisymmetric FR-structure, S be a non empty directed subset of R, and a be an element of R. If $a \in S$, then $a \leqslant \bigsqcup_R S$.

Let T be an up-complete non empty reflexive transitive antisymmetric FR-structure. One can check that every subset of T which is lower is also property(S).

One can prove the following propositions:

(22) For every finite up-complete non empty poset T holds every subset of T is inaccessible.

(23) Let R be a complete connected lattice, T be a Scott topological augmentation of R, and x be an element of T. Then $-{\downarrow}x$ is open.

(24) Let R be a complete connected lattice, T be a Scott topological augmentation of R, and S be a subset of T. Then S is open if and only if one of the following conditions is satisfied:

(i) $S =$ the carrier of T, or

(ii) $S \in \{-{\downarrow}x : x$ ranges over elements of $T\}$.

Let R be an up-complete non empty poset. One can check that there exists a correct topological augmentation of R which is order consistent.

Let us observe that there exists a top-lattice which is order consistent and complete.

The following three propositions are true:

(25) Let R be a non empty FR-structure and A be a subset of R. Suppose that for every element x of R holds ${\downarrow}x = \overline{\{x\}}$. If A is open, then A is upper.

(26) Let R be a non empty FR-structure and A be a subset of R. Suppose that for every element x of R holds ${\downarrow}x = \overline{\{x\}}$. Let A be a subset of R. If A is closed, then A is lower.

(27) For every up-complete inf-complete lattice T and for every net N in T and for every element i of N holds $\liminf(N{\restriction}i) = \liminf N$.

Let S be a non empty 1-sorted structure, let R be a non empty relational structure, and let f be a function from the carrier of R into the carrier of S. The functor $R * f$ yielding a strict non empty net structure over S is defined as follows:

(Def. 3) The relational structure of $R * f =$ the relational structure of R and the mapping of $R * f = f$.

Let S be a non empty 1-sorted structure, let R be a non empty transitive relational structure, and let f be a function from the carrier of R into the carrier of S. One can check that $R * f$ is transitive.

Let S be a non empty 1-sorted structure, let R be a non empty directed relational structure, and let f be a function from the carrier of R into the carrier of S. Note that $R * f$ is directed.

Let R be a non empty relational structure and let N be a prenet over R. The functor inf_net N yields a strict prenet over R and is defined by the condition (Def. 4).

(Def. 4) There exists a map f from N into R such that

(i) inf_net $N = N * f$, and

(ii) for every element i of the carrier of N holds $f(i) = \bigsqcap_R\{N(k); k$ ranges over elements of the carrier of N: $k \geqslant i\}$.

Let R be a non empty relational structure and let N be a net in R. One can verify that inf_net N is transitive.

Let R be a non empty relational structure and let N be a net in R. Note that inf_net N is directed.

Let R be an inf-complete non empty reflexive antisymmetric relational structure and let N be a net in R. One can verify that inf_net N is monotone.

Let R be an inf-complete non empty reflexive antisymmetric relational structure and let N be a net in R. One can verify that inf_net N is eventually-directed.

We now state several propositions:

(28) Let R be a non empty relational structure and N be a net in R. Then rng (the mapping of inf_net N) $= \{\bigsqcap_R\{N(i); i$ ranges over elements of the carrier of N: $i \geqslant j\}$: j ranges over elements of the carrier of $N\}$.

(29) For every up-complete inf-complete lattice R and for every net N in R holds sup inf_net $N = \liminf N$.

(30) For every up-complete inf-complete lattice R and for every net N in R and for every element i of N holds sup inf_net $N = \liminf(N{\restriction}i)$.

(31) Let R be an inf-complete semilattice, N be a net in R, and V be an upper subset of R. If inf_net N is eventually in V, then N is eventually in V.

(32) Let R be an inf-complete semilattice, N be a net in R, and V be a lower subset of R. If N is eventually in V, then inf_net N is eventually in V.

(33) Let R be a topological space-like order consistent up-complete inf-complete non empty top-lattice, N be a net in R, and x be an element of R. If $x \leqslant \liminf N$, then x is a cluster point of N.

(34) Let R be an order consistent up-complete inf-complete topological space-like non empty top-lattice, N be an eventually-directed net in R, and x be an element of R. Then $x \leqslant \liminf N$ if and only if x is a cluster point of N.

Acknowledgments

I would like to thank Dr. Grzegorz Bancerek for his help in the preparation of this article.

References

[1] Grzegorz Bancerek. Complete lattices. *Formalized Mathematics*, 2(**5**):719–725, 1991.

[2] Grzegorz Bancerek. Bounds in posets and relational substructures. *Formalized Mathematics*, 6(**1**):81–91, 1997.

[3] Grzegorz Bancerek. Directed sets, nets, ideals, filters, and maps. *Formalized Mathematics*, 6(**1**):93–107, 1997.

[4] Grzegorz Bancerek. Bases and refinements of topologies. *Formalized Mathematics*, 7(**1**):35–43, 1998.

[5] G. Gierz, K.H. Hofmann, K. Keimel, J.D. Lawson, M. Mislove, and D.S. Scott. *A Compendium of Continuous Lattices*. Springer-Verlag, Berlin, Heidelberg, New York, 1980.

[6] Artur Korniłowicz. Meet–continuous lattices. *Formalized Mathematics*, 6(**1**):159–167, 1997.

[7] Artur Korniłowicz. On the topological properties of meet-continuous lattices. *Formalized Mathematics*, 6(**2**):269–277, 1997.

[8] Beata Padlewska. Locally connected spaces. *Formalized Mathematics*, 2(**1**):93–96, 1991.

[9] Beata Padlewska and Agata Darmochwał. Topological spaces and continuous functions. *Formalized Mathematics*, 1(**1**):223–230, 1990.

[10] Andrzej Trybulec. Moore-Smith convergence. *Formalized Mathematics*, 6(**2**):213–225, 1997.

[11] Andrzej Trybulec. Scott topology. *Formalized Mathematics*, 6(**2**):311–319, 1997.

[12] Wojciech A. Trybulec. Groups. *Formalized Mathematics*, 1(**5**):821–827, 1990.

[13] Wojciech A. Trybulec. Partially ordered sets. *Formalized Mathematics*, 1(**2**):313–319, 1990.

[14] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[15] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

[16] Mariusz Żynel and Adam Guzowski. T_0 topological spaces. *Formalized Mathematics*, 5(**1**):75–77, 1996.

Received May 23, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

On Segre's Product of Partial Line Spaces

Adam Naumowicz
University of Białystok

Summary. In this paper the concept of partial line spaces is presented. We also construct the Segre's product for a family of partial line spaces indexed by an arbitrary nonempty set.

MML Identifier: `PENCIL_1`.

The terminology and notation used in this paper have been introduced in the following articles: [16], [1], [2], [7], [14], [6], [13], [11], [9], [10], [8], [5], [17], [15], [12], [4], and [3].

1. Preliminaries

One can prove the following propositions:

(1) For all functions f, g such that $\prod f = \prod g$ holds if f is non-empty, then g is non-empty.

(2) For every set X holds $2 \subseteq \overline{\overline{X}}$ iff there exist sets x, y such that $x \in X$ and $y \in X$ and $x \neq y$.

(3) For every set X such that $2 \subseteq \overline{\overline{X}}$ and for every set x there exists a set y such that $y \in X$ and $x \neq y$.

(4) For every set X holds $2 \subseteq \overline{\overline{X}}$ iff X is non trivial.

(5) For every set X holds $3 \subseteq \overline{\overline{X}}$ iff there exist sets x, y, z such that $x \in X$ and $y \in X$ and $z \in X$ and $x \neq y$ and $x \neq z$ and $y \neq z$.

(6) For every set X such that $3 \subseteq \overline{\overline{X}}$ and for all sets x, y there exists a set z such that $z \in X$ and $x \neq z$ and $y \neq z$.

ISSN 1426–2630

2. Partial Line Spaces

Let S be a topological structure. A block of S is an element of the topology of S.

Let S be a topological structure and let x, y be points of S. We say that x, y are collinear if and only if:

(Def. 1) $x = y$ or there exists a block l of S such that $\{x, y\} \subseteq l$.

Let S be a topological structure and let T be a subset of the carrier of S. We say that T is closed under lines if and only if:

(Def. 2) For every block l of S such that $2 \subseteq \overline{\overline{l \cap T}}$ holds $l \subseteq T$.

We say that T is strong if and only if:

(Def. 3) For all points x, y of S such that $x \in T$ and $y \in T$ holds x, y are collinear.

Let S be a topological structure. We say that S is void if and only if:

(Def. 4) The topology of S is empty.

We say that S is degenerated if and only if:

(Def. 5) The carrier of S is a block of S.

We say that S has non trivial blocks if and only if:

(Def. 6) For every block k of S holds $2 \subseteq \overline{\overline{k}}$.

We say that S is identifying close blocks if and only if:

(Def. 7) For all blocks k, l of S such that $2 \subseteq \overline{\overline{k \cap l}}$ holds $k = l$.

We say that S is truly-partial if and only if:

(Def. 8) There exist points x, y of S such that x, y are not collinear.

We say that S has no isolated points if and only if:

(Def. 9) For every point x of S there exists a block l of S such that $x \in l$.

We say that S is connected if and only if the condition (Def. 10) is satisfied.

(Def. 10) Let x, y be points of S. Then there exists a finite sequence f of elements of the carrier of S such that

(i) $x = f(1)$,

(ii) $y = f(\operatorname{len} f)$, and

(iii) for every natural number i such that $1 \leqslant i$ and $i < \operatorname{len} f$ and for all points a, b of S such that $a = f(i)$ and $b = f(i+1)$ holds a, b are collinear.

We say that S is strongly connected if and only if the condition (Def. 11) is satisfied.

(Def. 11) Let x be a point of S and X be a subset of the carrier of S. Suppose X is closed under lines and strong. Then there exists a finite sequence f of elements of $2^{\text{the carrier of } S}$ such that

(i) $X = f(1)$,

(ii) $x \in f(\operatorname{len} f)$,

(iii) for every subset W of the carrier of S such that $W \in \operatorname{rng} f$ holds W is closed under lines and strong, and

(iv) for every natural number i such that $1 \leqslant i$ and $i < \operatorname{len} f$ holds $2 \subseteq \overline{\overline{f(i) \cap f(i+1)}}$.

One can prove the following propositions:

(7) Let X be a non empty set. Suppose $3 \subseteq \overline{\overline{X}}$. Let S be a topological structure. Suppose the carrier of $S = X$ and the topology of $S = \{L; L$ ranges over subsets of X: $2 = \overline{\overline{L}}\}$. Then S is non empty, non void, non degenerated, non truly-partial, and identifying close blocks and has non trivial blocks and no isolated points.

(8) Let X be a non empty set. Suppose $3 \subseteq \overline{\overline{X}}$. Let K be a subset of X. Suppose $\overline{\overline{K}} = 2$. Let S be a topological structure. Suppose the carrier of $S = X$ and the topology of $S = \{L; L$ ranges over subsets of X: $2 = \overline{\overline{L}}\} \setminus \{K\}$. Then S is non empty, non void, non degenerated, truly-partial, and identifying close blocks and has non trivial blocks and no isolated points.

One can verify that there exists a topological structure which is strict, non empty, non void, non degenerated, non truly-partial, and identifying close blocks and has non trivial blocks and no isolated points and there exists a topological structure which is strict, non empty, non void, non degenerated, truly-partial, and identifying close blocks and has non trivial blocks and no isolated points.

Let S be a non void topological structure. Note that the topology of S is non empty.

Let S be a topological structure with no isolated points and let x, y be points of S. Let us observe that x, y are collinear if and only if:

(Def. 12) There exists a block l of S such that $\{x, y\} \subseteq l$.

A PLS is a non empty non void non degenerated identifying close blocks topological structure with non trivial blocks.

Let F be a binary relation. We say that F is TopStruct-yielding if and only if:

(Def. 13) For every set x such that $x \in \operatorname{rng} F$ holds x is a topological structure.

Let us mention that every function which is TopStruct-yielding is also 1-sorted yielding.

Let I be a set. Observe that there exists a many sorted set indexed by I which is TopStruct-yielding.

Let us note that there exists a function which is TopStruct-yielding.

Let F be a binary relation. We say that F is non-void-yielding if and only if:

(Def. 14) For every topological structure S such that $S \in \operatorname{rng} F$ holds S is non void.

Let F be a TopStruct-yielding function. Let us observe that F is non-void-yielding if and only if:

(Def. 15) For every set i such that $i \in \operatorname{rng} F$ holds i is a non void topological structure.

Let F be a binary relation. We say that F is trivial-yielding if and only if:

(Def. 16) For every set S such that $S \in \operatorname{rng} F$ holds S is trivial.

Let F be a binary relation. We say that F is non-Trivial-yielding if and only if:

(Def. 17) For every 1-sorted structure S such that $S \in \operatorname{rng} F$ holds S is non trivial.

Let us observe that every binary relation which is non-Trivial-yielding is also nonempty.

Let F be a 1-sorted yielding function. Let us observe that F is non-Trivial-yielding if and only if:

(Def. 18) For every set i such that $i \in \operatorname{rng} F$ holds i is a non trivial 1-sorted structure.

Let I be a non empty set, let A be a TopStruct-yielding many sorted set indexed by I, and let j be an element of I. Then $A(j)$ is a topological structure.

Let F be a binary relation. We say that F is PLS-yielding if and only if:

(Def. 19) For every set x such that $x \in \operatorname{rng} F$ holds x is a PLS.

One can verify the following observations:

∗ every function which is PLS-yielding is also nonempty and TopStruct-yielding,

∗ every TopStruct-yielding function which is PLS-yielding is also non-void-yielding, and

∗ every TopStruct-yielding function which is PLS-yielding is also non-Trivial-yielding.

Let I be a set. One can check that there exists a many sorted set indexed by I which is PLS-yielding.

Let I be a non empty set, let A be a PLS-yielding many sorted set indexed by I, and let j be an element of I. Then $A(j)$ is a PLS.

Let I be a set and let A be a many sorted set indexed by I. We say that A is Segre-like if and only if:

(Def. 20) There exists an element i of I such that for every element j of I such that $i \neq j$ holds $A(j)$ is non empty and trivial.

Let I be a set and let A be a many sorted set indexed by I. Note that $\{A\}$ is trivial-yielding.

The following proposition is true

(9) Let I be a non empty set, A be a many sorted set indexed by I, i be an

element of I, and S be a non trivial set. Then $A +\cdot (i, S)$ is non trivial-yielding.

Let I be a non empty set and let A be a many sorted set indexed by I. Observe that $\{A\}$ is Segre-like.

We now state two propositions:

(10) For every non empty set I and for every many sorted set A indexed by I and for all sets i, S holds $\{A\} +\cdot (i, S)$ is Segre-like.

(11) Let I be a non empty set, A be a nonempty 1-sorted yielding many sorted set indexed by I, and B be an element of the support of A. Then $\{B\}$ is a many sorted subset indexed by the support of A.

Let I be a non empty set and let A be a nonempty 1-sorted yielding many sorted set indexed by I. One can check that there exists a many sorted subset indexed by the support of A which is Segre-like, trivial-yielding, and non-empty.

Let I be a non empty set and let A be a non-Trivial-yielding 1-sorted yielding many sorted set indexed by I. Note that there exists a many sorted subset indexed by the support of A which is Segre-like, non trivial-yielding, and non-empty.

Let I be a non empty set. Observe that there exists a many sorted set indexed by I which is Segre-like and non trivial-yielding.

Let I be a non empty set and let B be a Segre-like non trivial-yielding many sorted set indexed by I. The functor index(B) yielding an element of I is defined by:

(Def. 21) $B(\mathrm{index}(B))$ is non trivial.

Next we state the proposition

(12) Let I be a non empty set, A be a Segre-like non trivial-yielding many sorted set indexed by I, and i be an element of I. If $i \neq \mathrm{index}(A)$, then $A(i)$ is non empty and trivial.

Let I be a non empty set. Note that every many sorted set indexed by I which is Segre-like and non trivial-yielding is also non-empty.

One can prove the following proposition

(13) Let I be a non empty set and A be a many sorted set indexed by I. Then $2 \subseteq \overline{\overline{\prod A}}$ if and only if A is non-empty and non trivial-yielding.

Let I be a non empty set and let B be a Segre-like non trivial-yielding many sorted set indexed by I. Note that $\prod B$ is non trivial.

3. Segre's Product

Let I be a non empty set and let A be a nonempty TopStruct-yielding many sorted set indexed by I. The functor Segre_Blocks A yields a family of subsets of $\prod$ (the support of A) and is defined by the condition (Def. 22).

(Def. 22) Let x be a set. Then $x \in$ Segre_Blocks A if and only if there exists a Segre-like many sorted subset B indexed by the support of A such that $x = \prod B$ and there exists an element i of I such that $B(i)$ is a block of $A(i)$.

Let I be a non empty set and let A be a nonempty TopStruct-yielding many sorted set indexed by I. The functor Segre_Product A yielding a non empty topological structure is defined as follows:

(Def. 23) Segre_Product $A = \langle\prod$ (the support of A),Segre_Blocks $A\rangle$.

The following propositions are true:

(14) Let I be a non empty set and A be a nonempty TopStruct-yielding many sorted set indexed by I. Then every point of Segre_Product A is a many sorted set indexed by I.

(15) Let I be a non empty set and A be a nonempty TopStruct-yielding many sorted set indexed by I. If there exists an element i of I such that $A(i)$ is non void, then Segre_Product A is non void.

(16) Let I be a non empty set and A be a nonempty TopStruct-yielding many sorted set indexed by I. Suppose that for every element i of I holds $A(i)$ is non degenerated and there exists an element i of I such that $A(i)$ is non void. Then Segre_Product A is non degenerated.

(17) Let I be a non empty set and A be a nonempty TopStruct-yielding many sorted set indexed by I. Suppose that for every element i of I holds $A(i)$ has non trivial blocks and there exists an element i of I such that $A(i)$ is non void. Then Segre_Product A has non trivial blocks.

(18) Let I be a non empty set and A be a nonempty TopStruct-yielding many sorted set indexed by I. Suppose that for every element i of I holds $A(i)$ is identifying close blocks and has non trivial blocks and there exists an element i of I such that $A(i)$ is non void. Then Segre_Product A is identifying close blocks.

Let I be a non empty set and let A be a PLS-yielding many sorted set indexed by I. Then Segre_Product A is a PLS.

One can prove the following propositions:

(19) Let T be a topological structure and S be a subset of the carrier of T. If S is trivial, then S is strong and closed under lines.

(20) Let S be an identifying close blocks topological structure, l be a block of S, and L be a subset of the carrier of S. If $L = l$, then L is closed under lines.

(21) Let S be a topological structure, l be a block of S, and L be a subset of the carrier of S. If $L = l$, then L is strong.

(22) For every non void topological structure S holds Ω_S is closed under lines.

(23) Let I be a non empty set, A be a Segre-like non trivial-yielding many sorted set indexed by I, and x, y be many sorted sets indexed by I. If $x \in \prod A$ and $y \in \prod A$, then for every set i such that $i \neq \mathrm{index}(A)$ holds $x(i) = y(i)$.

(24) Let I be a non empty set, A be a PLS-yielding many sorted set indexed by I, and x be a set. Then x is a block of Segre_Product A if and only if there exists a Segre-like non trivial-yielding many sorted subset L indexed by the support of A such that $x = \prod L$ and $L(\mathrm{index}(L))$ is a block of $A(\mathrm{index}(L))$.

(25) Let I be a non empty set, A be a PLS-yielding many sorted set indexed by I, and P be a many sorted set indexed by I. Suppose P is a point of Segre_Product A. Let i be an element of I and p be a point of $A(i)$. Then $P +\cdot (i, p)$ is a point of Segre_Product A.

(26) Let I be a non empty set and A, B be Segre-like non trivial-yielding many sorted sets indexed by I. Suppose $2 \subseteq \overline{\overline{\prod A \cap \prod B}}$. Then $\mathrm{index}(A) = \mathrm{index}(B)$ and for every set i such that $i \neq \mathrm{index}(A)$ holds $A(i) = B(i)$.

(27) Let I be a non empty set, A be a Segre-like non trivial-yielding many sorted set indexed by I, and N be a non trivial set. Then $A +\cdot (\mathrm{index}(A), N)$ is Segre-like and non trivial-yielding.

(28) Let S be a non empty non void identifying close blocks topological structure with no isolated points. If S is strongly connected, then S is connected.

(29) Let I be a non empty set, A be a PLS-yielding many sorted set indexed by I, and S be a subset of the carrier of Segre_Product A. Then S is non trivial, strong, and closed under lines if and only if there exists a Segre-like non trivial-yielding many sorted subset B indexed by the support of A such that $S = \prod B$ and for every subset C of the carrier of $A(\mathrm{index}(B))$ such that $C = B(\mathrm{index}(B))$ holds C is strong and closed under lines.

References

[1] Grzegorz Bancerek. Cardinal numbers. *Formalized Mathematics*, 1(**2**):377–382, 1990.

[2] Grzegorz Bancerek. König's theorem. *Formalized Mathematics*, 1(**3**):589–593, 1990.

[3] Grzegorz Bancerek. The reflection theorem. *Formalized Mathematics*, 1(**5**):973–977, 1990.

[4] Grzegorz Bancerek. The "way-below" relation. *Formalized Mathematics*, 6(**1**):169–176, 1997.

[5] Józef Białas. Group and field definitions. *Formalized Mathematics*, 1(**3**):433–439, 1990.

[6] Ewa Burakowska. Subalgebras of many sorted algebra. Lattice of subalgebras. *Formalized Mathematics*, 5(**1**):47–54, 1996.

[7] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[8] Artur Korniłowicz. Some basic properties of many sorted sets. *Formalized Mathematics*, 5(**3**):395–399, 1996.

[9] Beata Madras. Product of family of universal algebras. *Formalized Mathematics*, 4(**1**):103–108, 1993.

[10] Beata Padlewska and Agata Darmochwał. Topological spaces and continuous functions. *Formalized Mathematics*, 1(**1**):223–230, 1990.

[11] Piotr Rudnicki and Andrzej Trybulec. Multivariate polynomials with arbitrary number of variables. *Formalized Mathematics*, 9(**1**):95–110, 2001.

[12] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[13] Andrzej Trybulec. Many-sorted sets. *Formalized Mathematics*, 4(**1**):15–22, 1993.

[14] Andrzej Trybulec. Many sorted algebras. *Formalized Mathematics*, 5(**1**):37–42, 1996.

[15] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[16] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

[17] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

Received May 29, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

The Evaluation of Polynomials

Robert Milewski
University of Białystok

MML Identifier: POLYNOM4.

The articles [11], [15], [12], [3], [2], [17], [4], [18], [1], [13], [14], [9], [6], [7], [19], [16], [20], [5], [8], and [10] provide the terminology and notation for this paper.

1. Preliminaries

The following propositions are true:

(1) For every natural number n holds $0 -' n = 0$.

(2) Let D be a set, p be a finite sequence of elements of D, and i be a natural number. If $i < \operatorname{len} p$, then $p\restriction(i+1) = (p\restriction i) \frown \langle p(i+1)\rangle$.

(3) Let D be a non empty set, p be a finite sequence of elements of D, and n be a natural number. If $1 \leqslant n$ and $n \leqslant \operatorname{len} p$, then $p = (p\restriction(n -' 1)) \frown \langle p(n)\rangle \frown (p_{\lfloor n})$.

(4) Let L be an add-associative right zeroed right complementable non empty loop structure and n be a natural number. Then $\sum(n \mapsto 0_L) = 0_L$.

2. About Polynomials

The following propositions are true:

(5) Let L be an add-associative right zeroed right complementable left distributive non empty double loop structure and p be a sequence of L. Then $\mathbf{0}.\,L * p = \mathbf{0}.\,L$.

(6) For every non empty zero structure L holds $\operatorname{len} \mathbf{0}.\,L = 0$.

ISSN 1426–2630

(7) For every non degenerated non empty multiplicative loop with zero structure L holds $\operatorname{len}\mathbf{1}.\,L = 1$.

(8) For every non empty zero structure L and for every Polynomial p of L such that $\operatorname{len} p = 0$ holds $p = \mathbf{0}.\,L$.

(9) Let L be a right zeroed non empty loop structure, p, q be Polynomials of L, and n be a natural number. If $n \geqslant \operatorname{len} p$ and $n \geqslant \operatorname{len} q$, then $n \geqslant \operatorname{len}(p + q)$.

(10) Let L be an add-associative right zeroed right complementable non empty loop structure and p, q be Polynomials of L. If $\operatorname{len} p \neq \operatorname{len} q$, then $\operatorname{len}(p + q) = \max(\operatorname{len} p, \operatorname{len} q)$.

(11) Let L be an add-associative right zeroed right complementable non empty loop structure and p be a Polynomial of L. Then $\operatorname{len}(-p) = \operatorname{len} p$.

(12) Let L be an add-associative right zeroed right complementable non empty loop structure, p, q be Polynomials of L, and n be a natural number. If $n \geqslant \operatorname{len} p$ and $n \geqslant \operatorname{len} q$, then $n \geqslant \operatorname{len}(p - q)$.

(13) Let L be an add-associative right zeroed right complementable distributive commutative associative left unital field-like non empty double loop structure and p, q be Polynomials of L. If $\operatorname{len} p > 0$ and $\operatorname{len} q > 0$, then $\operatorname{len}(p * q) = (\operatorname{len} p + \operatorname{len} q) - 1$.

3. Leading Monomials

Let L be a non empty zero structure and let p be a Polynomial of L. The functor Leading-Monomial p yielding a sequence of L is defined as follows:

(Def. 1) $(\text{Leading-Monomial}\, p)(\operatorname{len} p -' 1) = p(\operatorname{len} p -' 1)$ and for every natural number n such that $n \neq \operatorname{len} p -' 1$ holds $(\text{Leading-Monomial}\, p)(n) = 0_L$.

The following proposition is true

(14) For every non empty zero structure L and for every Polynomial p of L holds $\text{Leading-Monomial}\, p = \mathbf{0}.\,L \mathbin{+\cdot} (\operatorname{len} p -' 1, p(\operatorname{len} p -' 1))$.

Let L be a non empty zero structure and let p be a Polynomial of L. Observe that Leading-Monomial p is finite-Support.

We now state several propositions:

(15) For every non empty zero structure L and for every Polynomial p of L such that $\operatorname{len} p = 0$ holds $\text{Leading-Monomial}\, p = \mathbf{0}.\,L$.

(16) For every non empty zero structure L holds $\text{Leading-Monomial}\,\mathbf{0}.\,L = \mathbf{0}.\,L$.

(17) For every non degenerated non empty multiplicative loop with zero structure L holds $\text{Leading-Monomial}\,\mathbf{1}.\,L = \mathbf{1}.\,L$.

(18) For every non empty zero structure L and for every Polynomial p of L holds len Leading-Monomial $p = \operatorname{len} p$.

(19) Let L be an add-associative right zeroed right complementable non empty loop structure and p be a Polynomial of L. Suppose $\operatorname{len} p \neq 0$. Then there exists a Polynomial q of L such that $\operatorname{len} q < \operatorname{len} p$ and $p = q +$ Leading-Monomial p and for every natural number n such that $n < \operatorname{len} p - 1$ holds $q(n) = p(n)$.

4. Evaluation of Polynomials

Let L be a unital non empty double loop structure, let p be a Polynomial of L, and let x be an element of the carrier of L. The functor $\operatorname{eval}(p, x)$ yields an element of L and is defined by the condition (Def. 2).

(Def. 2) There exists a finite sequence F of elements of the carrier of L such that $\operatorname{eval}(p, x) = \sum F$ and $\operatorname{len} F = \operatorname{len} p$ and for every natural number n such that $n \in \operatorname{dom} F$ holds $F(n) = p(n -' 1) \cdot \operatorname{power}_L(x, n -' 1)$.

Next we state several propositions:

(20) For every unital non empty double loop structure L and for every element x of the carrier of L holds $\operatorname{eval}(\mathbf{0}.\, L, x) = 0_L$.

(21) Let L be a well unital add-associative right zeroed right complementable associative non degenerated non empty double loop structure and x be an element of the carrier of L. Then $\operatorname{eval}(\mathbf{1}.\, L, x) = \mathbf{1}_L$.

(22) Let L be an Abelian add-associative right zeroed right complementable unital left distributive non empty double loop structure, p, q be Polynomials of L, and x be an element of the carrier of L. Then $\operatorname{eval}(p + q, x) = \operatorname{eval}(p, x) + \operatorname{eval}(q, x)$.

(23) Let L be an Abelian add-associative right zeroed right complementable unital distributive non empty double loop structure, p be a Polynomial of L, and x be an element of the carrier of L. Then $\operatorname{eval}(-p, x) = -\operatorname{eval}(p, x)$.

(24) Let L be an Abelian add-associative right zeroed right complementable unital distributive non empty double loop structure, p, q be Polynomials of L, and x be an element of the carrier of L. Then $\operatorname{eval}(p - q, x) = \operatorname{eval}(p, x) - \operatorname{eval}(q, x)$.

(25) Let L be an add-associative right zeroed right complementable right zeroed distributive unital non empty double loop structure, p be a Polynomial of L, and x be an element of the carrier of L. Then $\operatorname{eval}(\text{Leading-Monomial}\, p, x) = p(\operatorname{len} p -' 1) \cdot \operatorname{power}_L(x, \operatorname{len} p -' 1)$.

(26) Let L be an add-associative right zeroed right complementable distributive commutative associative field-like left unital non degenerated non

empty double loop structure, p, q be Polynomials of L, and x be an element of the carrier of L. Then eval(Leading-Monomial $p * q, x$) = eval(Leading-Monomial p, x) $\cdot$ eval(q, x).

(27) Let L be a field, p, q be Polynomials of L, and x be an element of the carrier of L. Then eval($p * q, x$) = eval(p, x) $\cdot$ eval(q, x).

5. Evaluation Homomorphism

Let L be an add-associative right zeroed right complementable distributive unital non empty double loop structure and let x be an element of the carrier of L. The functor Polynom-Evaluation(L, x) yields a map from Polynom-Ring L into L and is defined by:

(Def. 3) For every Polynomial p of L holds (Polynom-Evaluation(L, x))(p) = eval(p, x).

Let L be an add-associative right zeroed right complementable distributive associative well unital non degenerated non empty double loop structure and let x be an element of the carrier of L. One can verify that Polynom-Evaluation(L, x) is unity-preserving.

Let L be an Abelian add-associative right zeroed right complementable distributive unital non empty double loop structure and let x be an element of the carrier of L. One can verify that Polynom-Evaluation(L, x) is additive.

Let L be a field and let x be an element of the carrier of L. Observe that Polynom-Evaluation(L, x) is multiplicative.

Let L be a field and let x be an element of the carrier of L. Note that Polynom-Evaluation(L, x) is ring homomorphism.

References

[1] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[2] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[3] Czesław Byliński. Binary operations. *Formalized Mathematics*, 1(**1**):175–180, 1990.

[4] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[5] Agata Darmochwał and Yatsuka Nakamura. The topological space $\mathcal{E}^2_{\mathrm{T}}$. Arcs, line segments and special polygonal arcs. *Formalized Mathematics*, 2(**5**):617–621, 1991.

[6] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[7] Jarosław Kotowicz. Functions and finite sequences of real numbers. *Formalized Mathematics*, 3(**2**):275–278, 1992.

[8] Eugeniusz Kusak, Wojciech Leończuk, and Michał Muzalewski. Abelian groups, fields and vector spaces. *Formalized Mathematics*, 1(**2**):335–342, 1990.

[9] Robert Milewski. The ring of polynomials. *Formalized Mathematics*, 9(**2**):339–346, 2001.

[10] Michał Muzalewski. Construction of rings and left-, right-, and bi-modules over a ring. *Formalized Mathematics*, 2(**1**):3–11, 1991.

[11] Michał Muzalewski and Lesław W. Szczerba. Construction of finite sequences over ring and left-, right-, and bi-modules over a ring. *Formalized Mathematics*, 2(**1**):97–104, 1991.

[12] Takaya Nishiyama and Yasuho Mizuhara. Binary arithmetics. *Formalized Mathematics*, 4(**1**):83–86, 1993.

[13] Jan Popiołek. Real normed space. *Formalized Mathematics*, 2(**1**):111–115, 1991.

[14] Piotr Rudnicki and Andrzej Trybulec. Multivariate polynomials with arbitrary number of variables. *Formalized Mathematics*, 9(**1**):95–110, 2001.

[15] Wojciech Skaba and Michał Muzalewski. From double loops to fields. *Formalized Mathematics*, 2(**1**):185–191, 1991.

[16] Andrzej Trybulec and Czesław Byliński. Some properties of real numbers. *Formalized Mathematics*, 1(**3**):445–449, 1990.

[17] Wojciech A. Trybulec. Binary operations on finite sequences. *Formalized Mathematics*, 1(**5**):979–981, 1990.

[18] Wojciech A. Trybulec. Groups. *Formalized Mathematics*, 1(**5**):821–827, 1990.

[19] Wojciech A. Trybulec. Vectors in real linear space. *Formalized Mathematics*, 1(**2**):291–296, 1990.

[20] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

Received June 7, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

The Construction and Computation of While-Loop Programs for SCMPDS[1]

Jing-Chao Chen
Shanghai Jiaotong University

Summary. This article defines two while-loop statements on SCMPDS, i.e. "while<0" and "while>0", which resemble the while-statements of the common high language such as C. We previously presented a number of tricks for computing while-loop statements on SCMFSA, e.g. step-while. However, after inspecting a few realistic examples, we found that they are neither very useful nor of generalization. To cover much more computation cases of while-loop statements, we generalize the computation model of while-loop statements, based on the principle of Hoare's axioms on the verification of programs.

MML Identifier: SCMPDS_8.

The notation and terminology used here are introduced in the following articles: [14], [15], [19], [16], [1], [3], [17], [4], [5], [20], [2], [12], [13], [22], [23], [10], [6], [9], [7], [8], [11], [21], and [18].

1. Preliminaries

In this paper x, a denote Int positions and s denotes a state of SCMPDS.

We now state the proposition

(1) For every Int position a there exists a natural number i such that $a = \mathrm{intpos}\, i$.

Let t be a state of SCMPDS. The functor Dstate t yielding a state of SCMPDS is defined by the condition (Def. 1).

[1] This research is partially supported by the National Natural Science Foundation of China Grant No. 69873033.

(Def. 1) Let x be a set. Then

(i) if $x \in \text{Data-Loc}_{\text{SCM}}$, then $(\text{Dstate}\,t)(x) = t(x)$,

(ii) if $x \in$ the instruction locations of SCMPDS, then $(\text{Dstate}\,t)(x) =$ goto 0, and

(iii) if $x = \mathbf{IC}_{\text{SCMPDS}}$, then $(\text{Dstate}\,t)(x) = \text{inspos}\,0$.

One can prove the following four propositions:

(2) For all states t_1, t_2 of SCMPDS such that $t_1{\restriction}\text{Data-Loc}_{\text{SCM}} = t_2{\restriction}\text{Data-Loc}_{\text{SCM}}$ holds $\text{Dstate}\,t_1 = \text{Dstate}\,t_2$.

(3) For every state t of SCMPDS and for every instruction i of SCMPDS such that $\text{InsCode}(i) \in \{0, 4, 5, 6\}$ holds $\text{Dstate}\,t = \text{Dstate}\,\text{Exec}(i, t)$.

(4) $(\text{Dstate}\,s)(a) = s(a)$.

(5) Let a be an Int position. Then there exists a function f from $\prod$(the object kind of SCMPDS) into $\mathbb{N}$ such that for every state s of SCMPDS holds

(i) if $s(a) \leqslant 0$, then $f(s) = 0$, and

(ii) if $s(a) > 0$, then $f(s) = s(a)$.

2. The Construction and Several Basic Properties of "while<0" Program

Let a be an Int position, let i be an integer, and let I be a Program-block. The functor while $< 0(a, i, I)$ yielding a Program-block is defined by:

(Def. 2) while $< 0(a, i, I) = ((a, i) >= 0\_\text{goto}\,\text{card}\,I + 2)$; I; goto $(-(\text{card}\,I + 1))$.

Let I be a shiftable Program-block, let a be an Int position, and let i be an integer. Observe that while $< 0(a, i, I)$ is shiftable.

Let I be a No-StopCode Program-block, let a be an Int position, and let i be an integer. Note that while $< 0(a, i, I)$ is No-StopCode.

Next we state several propositions:

(6) For every Int position a and for every integer i and for every Program-block I holds card while $< 0(a, i, I) = \text{card}\,I + 2$.

(7) Let a be an Int position, i be an integer, m be a natural number, and I be a Program-block. Then $m < \text{card}\,I + 2$ if and only if $\text{inspos}\,m \in$ dom while $< 0(a, i, I)$.

(8) Let a be an Int position, i be an integer, and I be a Program-block. Then (while $< 0(a, i, I))(\text{inspos}\,0) = (a, i) >= 0\_\text{goto}\,\text{card}\,I + 2$ and (while $< 0(a, i, I))(\text{inspos}\,\text{card}\,I + 1) =$ goto $(-(\text{card}\,I + 1))$.

(9) Let s be a state of SCMPDS, I be a Program-block, a be an Int position, and i be an integer. If $s(\text{DataLoc}(s(a), i)) \geqslant 0$, then while $< 0(a, i, I)$ is closed on s and while $< 0(a, i, I)$ is halting on s.

(10) Let s be a state of SCMPDS, I be a Program-block, a, c be Int positions, and i be an integer. If $s(\mathrm{DataLoc}(s(a), i)) \geqslant 0$, then $\mathrm{IExec}(\text{while} < 0(a, i, I), s) = s +\cdot \text{Start-At}(\text{inspos}\,\text{card}\, I + 2)$.

(11) Let s be a state of SCMPDS, I be a Program-block, a be an Int position, and i be an integer. If $s(\mathrm{DataLoc}(s(a), i)) \geqslant 0$, then $\mathbf{IC}_{\mathrm{IExec}(\text{while}<0(a,i,I),s)} = \text{inspos}\,\text{card}\, I + 2$.

(12) Let s be a state of SCMPDS, I be a Program-block, a, b be Int positions, and i be an integer. If $s(\mathrm{DataLoc}(s(a), i)) \geqslant 0$, then $(\mathrm{IExec}(\text{while} < 0(a, i, I), s))(b) = s(b)$.

In this article we present several logical schemes. The scheme *WhileLHalt* deals with a unary functor $\mathcal{F}$ yielding a natural number, a state $\mathcal{A}$ of SCMPDS, a No-StopCode shiftable Program-block $\mathcal{B}$, an Int position $\mathcal{C}$, an integer $\mathcal{D}$, and a unary predicate $\mathcal{P}$, and states that:

$\mathcal{F}(\mathcal{A}) = \mathcal{F}(\mathcal{A})$ or $\mathcal{P}[\mathcal{A}]$ but while $< 0(\mathcal{C}, \mathcal{D}, \mathcal{B})$ is closed on $\mathcal{A}$ but while $< 0(\mathcal{C}, \mathcal{D}, \mathcal{B})$ is halting on $\mathcal{A}$

provided the following conditions are met:

- $\text{card}\, \mathcal{B} > 0$,
- For every state t of SCMPDS such that $\mathcal{P}[\text{Dstate}\, t]$ and $\mathcal{F}(\text{Dstate}\, t) = 0$ holds $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) \geqslant 0$,
- $\mathcal{P}[\text{Dstate}\, \mathcal{A}]$, and
- Let t be a state of SCMPDS. Suppose $\mathcal{P}[\text{Dstate}\, t]$ and $t(\mathcal{C}) = \mathcal{A}(\mathcal{C})$ and $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) < 0$. Then $(\mathrm{IExec}(\mathcal{B}, t))(\mathcal{C}) = t(\mathcal{C})$ and $\mathcal{B}$ is closed on t and $\mathcal{B}$ is halting on t and $\mathcal{F}(\text{Dstate}\, \mathrm{IExec}(\mathcal{B}, t)) < \mathcal{F}(\text{Dstate}\, t)$ and $\mathcal{P}[\text{Dstate}\, \mathrm{IExec}(\mathcal{B}, t)]$.

The scheme *WhileLExec* deals with a unary functor $\mathcal{F}$ yielding a natural number, a state $\mathcal{A}$ of SCMPDS, a No-StopCode shiftable Program-block $\mathcal{B}$, an Int position $\mathcal{C}$, an integer $\mathcal{D}$, and a unary predicate $\mathcal{P}$, and states that:

$\mathcal{F}(\mathcal{A}) = \mathcal{F}(\mathcal{A})$ or $\mathcal{P}[\mathcal{A}]$ but $\mathrm{IExec}(\text{while} < 0(\mathcal{C}, \mathcal{D}, \mathcal{B}), \mathcal{A}) = \mathrm{IExec}(\text{while} < 0(\mathcal{C}, \mathcal{D}, \mathcal{B}), \mathrm{IExec}(\mathcal{B}, \mathcal{A}))$

provided the parameters meet the following conditions:

- $\text{card}\, \mathcal{B} > 0$,
- $\mathcal{A}(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) < 0$,
- For every state t of SCMPDS such that $\mathcal{P}[\text{Dstate}\, t]$ and $\mathcal{F}(\text{Dstate}\, t) = 0$ holds $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) \geqslant 0$,
- $\mathcal{P}[\text{Dstate}\, \mathcal{A}]$, and
- Let t be a state of SCMPDS. Suppose $\mathcal{P}[\text{Dstate}\, t]$ and $t(\mathcal{C}) = \mathcal{A}(\mathcal{C})$ and $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) < 0$. Then $(\mathrm{IExec}(\mathcal{B}, t))(\mathcal{C}) = t(\mathcal{C})$ and $\mathcal{B}$ is closed on t and $\mathcal{B}$ is halting on t and $\mathcal{F}(\text{Dstate}\, \mathrm{IExec}(\mathcal{B}, t)) < \mathcal{F}(\text{Dstate}\, t)$ and $\mathcal{P}[\text{Dstate}\, \mathrm{IExec}(\mathcal{B}, t)]$.

One can prove the following propositions:

(13) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a be an Int position, i be an integer, X be a set, and f be a function from $\prod$ (the object kind of SCMPDS) into $\mathbb{N}$. Suppose that

(i) $\operatorname{card} I > 0$,

(ii) for every state t of SCMPDS such that $f(\text{Dstate}\, t) = 0$ holds $t(\text{DataLoc}(s(a), i)) \geqslant 0$, and

(iii) for every state t of SCMPDS such that for every Int position x such that $x \in X$ holds $t(x) = s(x)$ and $t(a) = s(a)$ and $t(\text{DataLoc}(s(a), i)) < 0$ holds $(\text{IExec}(I, t))(a) = t(a)$ and $f(\text{Dstate}\,\text{IExec}(I, t)) < f(\text{Dstate}\, t)$ and I is closed on t and halting on t and for every Int position x such that $x \in X$ holds $(\text{IExec}(I, t))(x) = t(x)$.

Then while $< 0(a, i, I)$ is closed on s and while $< 0(a, i, I)$ is halting on s.

(14) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a be an Int position, i be an integer, X be a set, and f be a function from $\prod$ (the object kind of SCMPDS) into $\mathbb{N}$. Suppose that

(i) $\operatorname{card} I > 0$,

(ii) $s(\text{DataLoc}(s(a), i)) < 0$,

(iii) for every state t of SCMPDS such that $f(\text{Dstate}\, t) = 0$ holds $t(\text{DataLoc}(s(a), i)) \geqslant 0$, and

(iv) for every state t of SCMPDS such that for every Int position x such that $x \in X$ holds $t(x) = s(x)$ and $t(a) = s(a)$ and $t(\text{DataLoc}(s(a), i)) < 0$ holds $(\text{IExec}(I, t))(a) = t(a)$ and I is closed on t and halting on t and $f(\text{Dstate}\,\text{IExec}(I, t)) < f(\text{Dstate}\, t)$ and for every Int position x such that $x \in X$ holds $(\text{IExec}(I, t))(x) = t(x)$.

Then $\text{IExec}(\text{while} < 0(a, i, I), s) = \text{IExec}(\text{while} < 0(a, i, I), \text{IExec}(I, s))$.

(15) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a be an Int position, i be an integer, and X be a set. Suppose that

(i) $\operatorname{card} I > 0$, and

(ii) for every state t of SCMPDS such that for every Int position x such that $x \in X$ holds $t(x) = s(x)$ and $t(a) = s(a)$ and $t(\text{DataLoc}(s(a), i)) < 0$ holds $(\text{IExec}(I, t))(a) = t(a)$ and $(\text{IExec}(I, t))(\text{DataLoc}(s(a), i)) > t(\text{DataLoc}(s(a), i))$ and I is closed on t and halting on t and for every Int position x such that $x \in X$ holds $(\text{IExec}(I, t))(x) = t(x)$.

Then while $< 0(a, i, I)$ is closed on s and while $< 0(a, i, I)$ is halting on s.

(16) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a be an Int position, i be an integer, and X be a set. Suppose that

(i) $s(\text{DataLoc}(s(a), i)) < 0$,

(ii) $\operatorname{card} I > 0$, and

(iii) for every state t of SCMPDS such that for every Int position x such

that $x \in X$ holds $t(x) = s(x)$ and $t(a) = s(a)$ and $t(\mathrm{DataLoc}(s(a), i)) < 0$ holds $(\mathrm{IExec}(I, t))(a) = t(a)$ and $(\mathrm{IExec}(I, t))(\mathrm{DataLoc}(s(a), i)) > t(\mathrm{DataLoc}(s(a), i))$ and I is closed on t and halting on t and for every Int position x such that $x \in X$ holds $(\mathrm{IExec}(I, t))(x) = t(x)$.
Then $\mathrm{IExec}(\mathrm{while} < 0(a, i, I), s) = \mathrm{IExec}(\mathrm{while} < 0(a, i, I), \mathrm{IExec}(I, s))$.

3. The Construction and Several Basic Properties of “while>0” Program

Let a be an Int position, let i be an integer, and let I be a Program-block. The functor $\mathrm{while} > 0(a, i, I)$ yields a Program-block and is defined by:

(Def. 3) $\mathrm{while} > 0(a, i, I) = ((a, i) <= 0\_\mathrm{goto}\,\mathrm{card}\, I + 2); I; \mathrm{goto}\ (-(\mathrm{card}\, I + 1))$.

Let I be a shiftable Program-block, let a be an Int position, and let i be an integer. One can verify that $\mathrm{while} > 0(a, i, I)$ is shiftable.

Let I be a No-StopCode Program-block, let a be an Int position, and let i be an integer. Note that $\mathrm{while} > 0(a, i, I)$ is No-StopCode.

Next we state several propositions:

(17) For every Int position a and for every integer i and for every Program-block I holds $\mathrm{card}\,\mathrm{while} > 0(a, i, I) = \mathrm{card}\, I + 2$.

(18) Let a be an Int position, i be an integer, m be a natural number, and I be a Program-block. Then $m < \mathrm{card}\, I + 2$ if and only if $\mathrm{inspos}\, m \in \mathrm{dom}\,\mathrm{while} > 0(a, i, I)$.

(19) Let a be an Int position, i be an integer, and I be a Program-block. Then $(\mathrm{while} > 0(a, i, I))(\mathrm{inspos}\, 0) = (a, i) <= 0\_\mathrm{goto}\,\mathrm{card}\, I + 2$ and $(\mathrm{while} > 0(a, i, I))(\mathrm{inspos}\,\mathrm{card}\, I + 1) = \mathrm{goto}\ (-(\mathrm{card}\, I + 1))$.

(20) Let s be a state of SCMPDS, I be a Program-block, a be an Int position, and i be an integer. If $s(\mathrm{DataLoc}(s(a), i)) \leqslant 0$, then $\mathrm{while} > 0(a, i, I)$ is closed on s and $\mathrm{while} > 0(a, i, I)$ is halting on s.

(21) Let s be a state of SCMPDS, I be a Program-block, a, c be Int positions, and i be an integer. If $s(\mathrm{DataLoc}(s(a), i)) \leqslant 0$, then $\mathrm{IExec}(\mathrm{while} > 0(a, i, I), s) = s {+\cdot}\, \mathrm{Start\text{-}At}(\mathrm{inspos}\,\mathrm{card}\, I + 2)$.

(22) Let s be a state of SCMPDS, I be a Program-block, a be an Int position, and i be an integer. If $s(\mathrm{DataLoc}(s(a), i)) \leqslant 0$, then $\mathbf{IC}_{\mathrm{IExec}(\mathrm{while}>0(a,i,I),s)} = \mathrm{inspos}\,\mathrm{card}\, I + 2$.

(23) Let s be a state of SCMPDS, I be a Program-block, a, b be Int positions, and i be an integer. If $s(\mathrm{DataLoc}(s(a), i)) \leqslant 0$, then $(\mathrm{IExec}(\mathrm{while} > 0(a, i, I), s))(b) = s(b)$.

Now we present two schemes. The scheme *WhileGHalt* deals with a unary functor $\mathcal{F}$ yielding a natural number, a state $\mathcal{A}$ of SCMPDS, a No-StopCode shiftable Program-block $\mathcal{B}$, an Int position $\mathcal{C}$, an integer $\mathcal{D}$, and a unary predicate $\mathcal{P}$, and states that:

$\mathcal{F}(\mathcal{A}) = \mathcal{F}(\mathcal{A})$ or $\mathcal{P}[\mathcal{A}]$ but while $> 0(\mathcal{C}, \mathcal{D}, \mathcal{B})$ is closed on $\mathcal{A}$ but while $> 0(\mathcal{C}, \mathcal{D}, \mathcal{B})$ is halting on $\mathcal{A}$

provided the parameters meet the following conditions:

- $\operatorname{card} \mathcal{B} > 0$,
- For every state t of SCMPDS such that $\mathcal{P}[\text{Dstate}\, t]$ and $\mathcal{F}(\text{Dstate}\, t) = 0$ holds $t(\text{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) \leqslant 0$,
- $\mathcal{P}[\text{Dstate}\, \mathcal{A}]$, and
- Let t be a state of SCMPDS. Suppose $\mathcal{P}[\text{Dstate}\, t]$ and $t(\mathcal{C}) = \mathcal{A}(\mathcal{C})$ and $t(\text{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) > 0$. Then $(\text{IExec}(\mathcal{B}, t))(\mathcal{C}) = t(\mathcal{C})$ and $\mathcal{B}$ is closed on t and $\mathcal{B}$ is halting on t and $\mathcal{F}(\text{Dstate}\, \text{IExec}(\mathcal{B}, t)) < \mathcal{F}(\text{Dstate}\, t)$ and $\mathcal{P}[\text{Dstate}\, \text{IExec}(\mathcal{B}, t)]$.

The scheme *WhileGExec* deals with a unary functor $\mathcal{F}$ yielding a natural number, a state $\mathcal{A}$ of SCMPDS, a No-StopCode shiftable Program-block $\mathcal{B}$, an Int position $\mathcal{C}$, an integer $\mathcal{D}$, and a unary predicate $\mathcal{P}$, and states that:

$\mathcal{F}(\mathcal{A}) = \mathcal{F}(\mathcal{A})$ or $\mathcal{P}[\mathcal{A}]$ but IExec(while $> 0(\mathcal{C}, \mathcal{D}, \mathcal{B}), \mathcal{A}) =$ IExec(while $> 0(\mathcal{C}, \mathcal{D}, \mathcal{B}), \text{IExec}(\mathcal{B}, \mathcal{A}))$

provided the following conditions are satisfied:

- $\operatorname{card} \mathcal{B} > 0$,
- $\mathcal{A}(\text{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) > 0$,
- For every state t of SCMPDS such that $\mathcal{P}[\text{Dstate}\, t]$ and $\mathcal{F}(\text{Dstate}\, t) = 0$ holds $t(\text{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) \leqslant 0$,
- $\mathcal{P}[\text{Dstate}\, \mathcal{A}]$, and
- Let t be a state of SCMPDS. Suppose $\mathcal{P}[\text{Dstate}\, t]$ and $t(\mathcal{C}) = \mathcal{A}(\mathcal{C})$ and $t(\text{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) > 0$. Then $(\text{IExec}(\mathcal{B}, t))(\mathcal{C}) = t(\mathcal{C})$ and $\mathcal{B}$ is closed on t and $\mathcal{B}$ is halting on t and $\mathcal{F}(\text{Dstate}\, \text{IExec}(\mathcal{B}, t)) < \mathcal{F}(\text{Dstate}\, t)$ and $\mathcal{P}[\text{Dstate}\, \text{IExec}(\mathcal{B}, t)]$.

One can prove the following propositions:

(24) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a be an Int position, i, c be integers, X, Y be sets, and f be a function from $\prod$(the object kind of SCMPDS) into $\mathbb{N}$. Suppose that

(i) $\operatorname{card} I > 0$,

(ii) for every state t of SCMPDS such that $f(\text{Dstate}\, t) = 0$ holds $t(\text{DataLoc}(s(a), i)) \leqslant 0$,

(iii) for every x such that $x \in X$ holds $s(x) \geqslant c + s(\text{DataLoc}(s(a), i))$, and

(iv) for every state t of SCMPDS such that for every x such that $x \in X$ holds $t(x) \geqslant c + t(\text{DataLoc}(s(a), i))$ and for every x such that $x \in Y$ holds $t(x) = s(x)$ and $t(a) = s(a)$ and $t(\text{DataLoc}(s(a), i)) > 0$ holds $(\text{IExec}(I, t))(a) = t(a)$ and I is closed on t and halting on t and $f(\text{Dstate}\, \text{IExec}(I, t)) < f(\text{Dstate}\, t)$ and for every x such that $x \in X$ holds $(\text{IExec}(I, t))(x) \geqslant$

$c + (\text{IExec}(I, t))(\text{DataLoc}(s(a), i))$ and for every x such that $x \in Y$ holds $(\text{IExec}(I, t))(x) = t(x)$.

Then while $> 0(a, i, I)$ is closed on s and while $> 0(a, i, I)$ is halting on s.

(25) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a be an Int position, i, c be integers, X, Y be sets, and f be a function from $\prod$(the object kind of SCMPDS) into $\mathbb{N}$. Suppose that

(i) $s(\text{DataLoc}(s(a), i)) > 0$,

(ii) $\text{card}\, I > 0$,

(iii) for every state t of SCMPDS such that $f(\text{Dstate}\, t) = 0$ holds $t(\text{DataLoc}(s(a), i)) \leqslant 0$,

(iv) for every x such that $x \in X$ holds $s(x) \geqslant c + s(\text{DataLoc}(s(a), i))$, and

(v) for every state t of SCMPDS such that for every x such that $x \in X$ holds $t(x) \geqslant c + t(\text{DataLoc}(s(a), i))$ and for every x such that $x \in Y$ holds $t(x) = s(x)$ and $t(a) = s(a)$ and $t(\text{DataLoc}(s(a), i)) > 0$ holds $(\text{IExec}(I, t))(a) = t(a)$ and I is closed on t and halting on t and $f(\text{Dstate}\,\text{IExec}(I, t)) < f(\text{Dstate}\, t)$ and for every x such that $x \in X$ holds $(\text{IExec}(I, t))(x) \geqslant c + (\text{IExec}(I, t))(\text{DataLoc}(s(a), i))$ and for every x such that $x \in Y$ holds $(\text{IExec}(I, t))(x) = t(x)$.

Then $\text{IExec}(\text{while} > 0(a, i, I), s) = \text{IExec}(\text{while} > 0(a, i, I), \text{IExec}(I, s))$.

(26) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a be an Int position, i be an integer, X be a set, and f be a function from $\prod$(the object kind of SCMPDS) into $\mathbb{N}$. Suppose that

(i) $\text{card}\, I > 0$,

(ii) for every state t of SCMPDS such that $f(\text{Dstate}\, t) = 0$ holds $t(\text{DataLoc}(s(a), i)) \leqslant 0$, and

(iii) for every state t of SCMPDS such that for every x such that $x \in X$ holds $t(x) = s(x)$ and $t(a) = s(a)$ and $t(\text{DataLoc}(s(a), i)) > 0$ holds $(\text{IExec}(I, t))(a) = t(a)$ and I is closed on t and halting on t and $f(\text{Dstate}\,\text{IExec}(I, t)) < f(\text{Dstate}\, t)$ and for every x such that $x \in X$ holds $(\text{IExec}(I, t))(x) = t(x)$.

Then while $> 0(a, i, I)$ is closed on s and while $> 0(a, i, I)$ is halting on s and if $s(\text{DataLoc}(s(a), i)) > 0$, then $\text{IExec}(\text{while} > 0(a, i, I), s) = \text{IExec}(\text{while} > 0(a, i, I), \text{IExec}(I, s))$.

(27) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a be an Int position, i, c be integers, and X, Y be sets. Suppose that

(i) $\text{card}\, I > 0$,

(ii) for every x such that $x \in X$ holds $s(x) \geqslant c + s(\text{DataLoc}(s(a), i))$, and

(iii) for every state t of SCMPDS such that for every x such that $x \in X$ holds $t(x) \geqslant c + t(\text{DataLoc}(s(a), i))$ and for every x such that $x \in Y$ holds $t(x) = s(x)$ and $t(a) = s(a)$ and $t(\text{DataLoc}(s(a), i)) > 0$ holds

(IExec(I, t))(a) $=$ $t(a)$ and I is closed on t and halting on t and (IExec(I, t))(DataLoc($s(a), i$)) $<$ t(DataLoc($s(a), i$)) and for every x such that $x \in X$ holds (IExec(I, t))(x) $\geqslant c+$(IExec(I, t))(DataLoc($s(a), i$)) and for every x such that $x \in Y$ holds (IExec(I, t))(x) $= t(x)$.

Then while $> 0(a, i, I)$ is closed on s and while $> 0(a, i, I)$ is halting on s and if s(DataLoc($s(a), i$)) > 0, then IExec(while $> 0(a, i, I), s$) $=$ IExec(while $> 0(a, i, I)$, IExec(I, s)).

(28) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a be an Int position, i be an integer, and X be a set. Suppose that

(i) card $I > 0$, and

(ii) for every state t of SCMPDS such that for every x such that $x \in X$ holds $t(x) = s(x)$ and $t(a) = s(a)$ and t(DataLoc($s(a), i$)) > 0 holds (IExec(I, t))(a) $=$ $t(a)$ and I is closed on t and halting on t and (IExec(I, t))(DataLoc($s(a), i$)) $<$ t(DataLoc($s(a), i$)) and for every x such that $x \in X$ holds (IExec(I, t))(x) $= t(x)$.

Then while $> 0(a, i, I)$ is closed on s and while $> 0(a, i, I)$ is halting on s and if s(DataLoc($s(a), i$)) > 0, then IExec(while $> 0(a, i, I), s$) $=$ IExec(while $> 0(a, i, I)$, IExec(I, s)).

(29) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a be an Int position, i, c be integers, and X be a set. Suppose that

(i) card $I > 0$,

(ii) for every x such that $x \in X$ holds $s(x) \geqslant c + s$(DataLoc($s(a), i$)), and

(iii) for every state t of SCMPDS such that for every x such that $x \in X$ holds $t(x) \geqslant c + t$(DataLoc($s(a), i$)) and $t(a) = s(a)$ and t(DataLoc($s(a), i$)) > 0 holds (IExec(I, t))(a) $= t(a)$ and I is closed on t and halting on t and (IExec(I, t))(DataLoc($s(a), i$)) $<$ t(DataLoc($s(a), i$)) and for every x such that $x \in X$ holds (IExec(I, t))(x) $\geqslant c +$ (IExec(I, t))(DataLoc($s(a), i$)).

Then while $> 0(a, i, I)$ is closed on s and while $> 0(a, i, I)$ is halting on s and if s(DataLoc($s(a), i$)) > 0, then IExec(while $> 0(a, i, I), s$) $=$ IExec(while $> 0(a, i, I)$, IExec(I, s)).

References

[1] Grzegorz Bancerek. Cardinal numbers. *Formalized Mathematics*, 1(**2**):377–382, 1990.

[2] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[3] Grzegorz Bancerek. König's theorem. *Formalized Mathematics*, 1(**3**):589–593, 1990.

[4] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[5] Czesław Byliński. Functions from a set to a set. *Formalized Mathematics*, 1(**1**):153–164, 1990.

[6] Jing-Chao Chen. Computation and program shift in the SCMPDS computer. *Formalized Mathematics*, 8(**1**):193–199, 1999.

[7] Jing-Chao Chen. Computation of two consecutive program blocks for SCMPDS. *Formalized Mathematics*, 8(**1**):211–217, 1999.

[8] Jing-Chao Chen. The construction and computation of conditional statements for SCMPDS. *Formalized Mathematics*, 8(**1**):219–234, 1999.

[9] Jing-Chao Chen. The construction and shiftability of program blocks for SCMPDS. *Formalized Mathematics*, 8(**1**):201–210, 1999.

[10] Jing-Chao Chen. The SCMPDS computer and the basic semantics of its instructions. *Formalized Mathematics*, 8(**1**):183–191, 1999.

[11] Jing-Chao Chen. Recursive Euclide algorithm. *Formalized Mathematics*, 9(**1**):1–4, 2001.

[12] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[13] Krzysztof Hryniewiecki. Recursive definitions. *Formalized Mathematics*, 1(**2**):321–328, 1990.

[14] Yatsuka Nakamura and Andrzej Trybulec. A mathematical model of CPU. *Formalized Mathematics*, 3(**2**):151–160, 1992.

[15] Yatsuka Nakamura and Andrzej Trybulec. On a mathematical model of programs. *Formalized Mathematics*, 3(**2**):241–250, 1992.

[16] Yasushi Tanaka. On the decomposition of the states of SCM. *Formalized Mathematics*, 5(**1**):1–8, 1996.

[17] Andrzej Trybulec. Domains and their Cartesian products. *Formalized Mathematics*, 1(**1**):115–122, 1990.

[18] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[19] Andrzej Trybulec and Yatsuka Nakamura. Some remarks on the simple concrete model of computer. *Formalized Mathematics*, 4(**1**):51–56, 1993.

[20] Michał J. Trybulec. Integers. *Formalized Mathematics*, 1(**3**):501–505, 1990.

[21] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[22] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

[23] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

Received June 14, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Insert Sort on SCMPDS[1]

Jing-Chao Chen
Shanghai Jiaotong University

Summary. The goal of this article is to examine the effectiveness of "for-loop" and "while-loop" statements on SCMPDS by insert sort. In this article, first of all, we present an approach to compute the execution result of "for-loop" program by "loop-invariant", based on Hoare's axioms for program verification. Secondly, we extend the fundamental properties of the finite sequence and complex instructions of SCMPDS. Finally, we prove the correctness of the insert sort program described in the article.

MML Identifier: `SCPISORT`.

The terminology and notation used in this paper have been introduced in the following articles: [16], [19], [1], [3], [4], [20], [2], [13], [15], [9], [5], [8], [6], [7], [12], [10], [11], [17], [21], [18], and [14].

1. Preliminaries

In this paper n, p_0 are natural numbers.

Let f be a finite sequence of elements of $\mathbb{Z}$, let s be a state of SCMPDS, and let m be a natural number. We say that f is FinSequence on s, m if and only if:

(Def. 1) For every natural number i such that $1 \leqslant i$ and $i \leqslant \operatorname{len} f$ holds $f(i) = s(\operatorname{intpos} m + i)$.

We now state four propositions:

(1) Let f be a finite sequence of elements of $\mathbb{Z}$ and m, n be natural numbers. If $m \geqslant n$, then f is non decreasing on m, n.

[1] This research is partially supported by the National Natural Science Foundation of China Grant No. 69873033.

(2) Let s be a state of SCMPDS and n, m be natural numbers. Then there exists a finite sequence f of elements of $\mathbb{Z}$ such that $\operatorname{len} f = n$ and for every natural number i such that $1 \leqslant i$ and $i \leqslant \operatorname{len} f$ holds $f(i) = s(\operatorname{intpos} m+i)$.

(3) Let s be a state of SCMPDS and n, m be natural numbers. Then there exists a finite sequence f of elements of $\mathbb{Z}$ such that $\operatorname{len} f = n$ and f is FinSequence on s, m.

(4) Let f, g be finite sequences of elements of $\mathbb{Z}$ and m, n be natural numbers. Suppose that $1 \leqslant n$ and $n \leqslant \operatorname{len} f$ and $1 \leqslant m$ and $m \leqslant \operatorname{len} f$ and $\operatorname{len} f = \operatorname{len} g$ and $f(m) = g(n)$ and $f(n) = g(m)$ and for every natural number k such that $k \neq m$ and $k \neq n$ and $1 \leqslant k$ and $k \leqslant \operatorname{len} f$ holds $f(k) = g(k)$. Then f and g are fiberwise equipotent.

The following propositions are true:

(5) For all states s_1, s_2 of SCMPDS such that for every Int position a holds $s_1(a) = s_2(a)$ holds $\operatorname{Dstate} s_1 = \operatorname{Dstate} s_2$.

(6) Let s be a state of SCMPDS, I be a No-StopCode Program-block, and j be a parahalting shiftable instruction of SCMPDS. Suppose I is closed on s and halting on s. Then I; j is closed on s and I; j is halting on s.

(7) Let s be a state of SCMPDS, I be a No-StopCode Program-block, J be a shiftable parahalting Program-block, and a be an Int position. If I is closed on s and halting on s, then $(\operatorname{IExec}(I;\ J, s))(a) = (\operatorname{IExec}(J, \operatorname{IExec}(I, s)))(a)$.

(8) Let s be a state of SCMPDS, I be a No-StopCode parahalting Program-block, J be a shiftable Program-block, and a be an Int position. If J is closed on $\operatorname{IExec}(I, s)$ and halting on $\operatorname{IExec}(I, s)$, then $(\operatorname{IExec}(I;\ J, s))(a) = (\operatorname{IExec}(J, \operatorname{IExec}(I, s)))(a)$.

(9) Let s be a state of SCMPDS, I be a Program-block, and J be a shiftable parahalting Program-block. Suppose I is closed on s and halting on s. Then I; J is closed on s and I; J is halting on s.

(10) Let s be a state of SCMPDS, I be a parahalting Program-block, and J be a shiftable Program-block. Suppose J is closed on $\operatorname{IExec}(I, s)$ and halting on $\operatorname{IExec}(I, s)$. Then I; J is closed on s and I; J is halting on s.

(11) Let s be a state of SCMPDS, I be a Program-block, and j be a parahalting shiftable instruction of SCMPDS. Suppose I is closed on s and halting on s. Then I; j is closed on s and I; j is halting on s.

2. Computing the Execution Result of For-Loop Program by Loop-Invariant

In this article we present several logical schemes. The scheme *ForDownHalt* deals with a state $\mathcal{A}$ of SCMPDS, a No-StopCode shiftable Program-block $\mathcal{B}$, an Int position $\mathcal{C}$, an integer $\mathcal{D}$, a natural number $\mathcal{E}$, and a unary predicate $\mathcal{P}$, and states that:

> $\mathcal{P}[\mathcal{A}]$ or not $\mathcal{P}[\mathcal{A}]$ but for-down$(\mathcal{C}, \mathcal{D}, \mathcal{E}, \mathcal{B})$ is closed on $\mathcal{A}$ but for-down$(\mathcal{C}, \mathcal{D}, \mathcal{E}, \mathcal{B})$ is halting on $\mathcal{A}$

provided the following requirements are met:

- $\mathcal{E} > 0$,
- $\mathcal{P}[\mathrm{Dstate}\,\mathcal{A}]$, and
- Let t be a state of SCMPDS. Suppose $\mathcal{P}[\mathrm{Dstate}\,t]$ and $t(\mathcal{C}) = \mathcal{A}(\mathcal{C})$ and $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) > 0$. Then $(\mathrm{IExec}(\mathcal{B};\ \mathrm{AddTo}(\mathcal{C}, \mathcal{D}, -\mathcal{E}), t))(\mathcal{C}) = t(\mathcal{C})$ and $(\mathrm{IExec}(\mathcal{B};\ \mathrm{AddTo}(\mathcal{C}, \mathcal{D}, -\mathcal{E}), t))(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) = t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) - \mathcal{E}$ and $\mathcal{B}$ is closed on t and $\mathcal{B}$ is halting on t and $\mathcal{P}[\mathrm{Dstate}\,\mathrm{IExec}(\mathcal{B};\ \mathrm{AddTo}(\mathcal{C}, \mathcal{D}, -\mathcal{E}), t)]$.

The scheme *ForDownExec* deals with a state $\mathcal{A}$ of SCMPDS, a No-StopCode shiftable Program-block $\mathcal{B}$, an Int position $\mathcal{C}$, an integer $\mathcal{D}$, a natural number $\mathcal{E}$, and a unary predicate $\mathcal{P}$, and states that:

> $\mathcal{P}[\mathcal{A}]$ or not $\mathcal{P}[\mathcal{A}]$ but $\mathrm{IExec}(\text{for-down}(\mathcal{C}, \mathcal{D}, \mathcal{E}, \mathcal{B}), \mathcal{A}) = \mathrm{IExec}(\text{for-down}(\mathcal{C}, \mathcal{D}, \mathcal{E}, \mathcal{B}), \mathrm{IExec}(\mathcal{B};\ \mathrm{AddTo}(\mathcal{C}, \mathcal{D}, -\mathcal{E}), \mathcal{A}))$

provided the parameters meet the following conditions:

- $\mathcal{E} > 0$,
- $\mathcal{A}(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) > 0$,
- $\mathcal{P}[\mathrm{Dstate}\,\mathcal{A}]$, and
- Let t be a state of SCMPDS. Suppose $\mathcal{P}[\mathrm{Dstate}\,t]$ and $t(\mathcal{C}) = \mathcal{A}(\mathcal{C})$ and $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) > 0$. Then $(\mathrm{IExec}(\mathcal{B};\ \mathrm{AddTo}(\mathcal{C}, \mathcal{D}, -\mathcal{E}), t))(\mathcal{C}) = t(\mathcal{C})$ and $(\mathrm{IExec}(\mathcal{B};\ \mathrm{AddTo}(\mathcal{C}, \mathcal{D}, -\mathcal{E}), t))(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) = t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) - \mathcal{E}$ and $\mathcal{B}$ is closed on t and $\mathcal{B}$ is halting on t and $\mathcal{P}[\mathrm{Dstate}\,\mathrm{IExec}(\mathcal{B};\ \mathrm{AddTo}(\mathcal{C}, \mathcal{D}, -\mathcal{E}), t)]$.

The scheme *ForDownEnd* deals with a state $\mathcal{A}$ of SCMPDS, a No-StopCode shiftable Program-block $\mathcal{B}$, an Int position $\mathcal{C}$, an integer $\mathcal{D}$, a natural number $\mathcal{E}$, and a unary predicate $\mathcal{P}$, and states that:

> $\mathcal{P}[\mathcal{A}]$ or not $\mathcal{P}[\mathcal{A}]$ but $(\mathrm{IExec}(\text{for-down}(\mathcal{C}, \mathcal{D}, \mathcal{E}, \mathcal{B}), \mathcal{A}))(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) \leqslant 0$ but $\mathcal{P}[\mathrm{Dstate}\,\mathrm{IExec}(\text{for-down}(\mathcal{C}, \mathcal{D}, \mathcal{E}, \mathcal{B}), \mathcal{A})]$

provided the parameters have the following properties:

- $\mathcal{E} > 0$,
- $\mathcal{P}[\mathrm{Dstate}\,\mathcal{A}]$, and
- Let t be a state of SCMPDS. Suppose $\mathcal{P}[\mathrm{Dstate}\,t]$ and $t(\mathcal{C}) = \mathcal{A}(\mathcal{C})$ and $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) > 0$. Then $(\mathrm{IExec}(\mathcal{B};\ \mathrm{AddTo}(\mathcal{C}, \mathcal{D}, -\mathcal{E}), t))(\mathcal{C}) = t(\mathcal{C})$ and $(\mathrm{IExec}(\mathcal{B};\ \mathrm{AddTo}(\mathcal{C}, \mathcal{D}, -\mathcal{E}), t))(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) = t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) - \mathcal{E}$ and $\mathcal{B}$ is closed on t and $\mathcal{B}$ is halting on t and $\mathcal{P}[\mathrm{Dstate}\,\mathrm{IExec}(\mathcal{B};\ \mathrm{AddTo}(\mathcal{C}, \mathcal{D}, -\mathcal{E}), t)]$.

We now state three propositions:

(12) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a, x, y be Int positions, i, c be integers, and n be a natural number.

Suppose that

(i) $n > 0$,

(ii) $s(x) \geqslant s(y) + c$, and

(iii) for every state t of SCMPDS such that $t(x) \geqslant t(y) + c$ and $t(a) = s(a)$ and $t(\mathrm{DataLoc}(s(a), i)) > 0$ holds $(\mathrm{IExec}(I;\ \mathrm{AddTo}(a, i, -n), t))(a) = t(a)$ and $(\mathrm{IExec}(I;\ \mathrm{AddTo}(a, i, -n), t))(\mathrm{DataLoc}(s(a), i)) = t(\mathrm{DataLoc}(s(a), i)) - n$ and I is closed on t and halting on t and $(\mathrm{IExec}(I;\ \mathrm{AddTo}(a, i, -n), t))(x) \geqslant (\mathrm{IExec}(I;\ \mathrm{AddTo}(a, i, -n), t))(y) + c$.

Then for-down(a, i, n, I) is closed on s and for-down(a, i, n, I) is halting on s.

(13) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a, x, y be Int positions, i, c be integers, and n be a natural number. Suppose that

(i) $n > 0$,

(ii) $s(x) \geqslant s(y) + c$,

(iii) $s(\mathrm{DataLoc}(s(a), i)) > 0$, and

(iv) for every state t of SCMPDS such that $t(x) \geqslant t(y) + c$ and $t(a) = s(a)$ and $t(\mathrm{DataLoc}(s(a), i)) > 0$ holds $(\mathrm{IExec}(I;\ \mathrm{AddTo}(a, i, -n), t))(a) = t(a)$ and $(\mathrm{IExec}(I;\ \mathrm{AddTo}(a, i, -n), t))(\mathrm{DataLoc}(s(a), i)) = t(\mathrm{DataLoc}(s(a), i)) - n$ and I is closed on t and halting on t and $(\mathrm{IExec}(I;\ \mathrm{AddTo}(a, i, -n), t))(x) \geqslant (\mathrm{IExec}(I;\ \mathrm{AddTo}(a, i, -n), t))(y) + c$.

Then IExec(for-down$(a, i, n, I), s) =$ IExec(for-down(a, i, n, I), $\mathrm{IExec}(I;\ \mathrm{AddTo}(a, i, -n), s))$.

(14) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a be an Int position, i be an integer, and n be a natural number. Suppose that

(i) $s(\mathrm{DataLoc}(s(a), i)) > 0$,

(ii) $n > 0$,

(iii) $\mathrm{card}\, I > 0$,

(iv) $a \neq \mathrm{DataLoc}(s(a), i)$, and

(v) for every state t of SCMPDS such that $t(a) = s(a)$ holds $(\mathrm{IExec}(I, t))(a) = t(a)$ and $(\mathrm{IExec}(I, t))(\mathrm{DataLoc}(s(a), i)) = t(\mathrm{DataLoc}(s(a), i))$ and I is closed on t and halting on t.

Then for-down(a, i, n, I) is closed on s and for-down(a, i, n, I) is halting on s.

3. A Program for Insert Sort

Let n, p_0 be natural numbers. The functor insert-sort(n, p_0) yielding a Program-block is defined by the condition (Def. 2).

(Def. 2) insert-sort(n, p_0) = (GBP :=0); ((GBP)$_1$:=0); ((GBP)$_2$:=$n-1$); ((GBP)$_3$:=p_0); for-down(GBP, 2, 1, AddTo(GBP, 3, 1); ((GBP, 4) := (GBP, 3)); AddTo(GBP, 1, 1); ((GBP, 6) := (GBP, 1)); while $>$ 0(GBP, 6, ((GBP, 5) := (intpos 4, -1)); SubFrom(GBP, 5, intpos 4, 0); (**if** GBP $>$ 5 **then** ((GBP, 5) := (intpos 4, -1)); ((intpos 4, -1) := (intpos 4, 0)); ((intpos 4, 0) := (GBP, 5)); AddTo(GBP, 4, -1); AddTo(GBP, 6, -1) **else** Load((GBP)$_6$:=0)))).

4. The Property of Insert Sort and Its Correctness

We now state two propositions:

(15) card insert-sort$(n, p_0) = 23$.

(16) If $p_0 \geqslant 7$, then insert-sort(n, p_0) is parahalting.

One can prove the following propositions:

(17) Let s be a state of SCMPDS, f, g be finite sequences of elements of $\mathbb{Z}$, and k_0, k be natural numbers. Suppose that $s(a_4) \geqslant 7 + s(a_6)$ and $s(\mathrm{GBP}) = 0$ and $k = s(a_6)$ and $k_0 = s(a_4) - s(a_6) - 1$ and f is FinSequence on s, k_0 and g is FinSequence on IExec(I_2, s), k_0 and len f = len g and len $f > k$ and f is non decreasing on 1, k. Then

(i) f and g are fiberwise equipotent,

(ii) g is non decreasing on 1, $k+1$,

(iii) for every natural number i such that $i > k+1$ and $i \leqslant \mathrm{len}\, f$ holds $f(i) = g(i)$, and

(iv) for every natural number i such that $1 \leqslant i$ and $i \leqslant k+1$ there exists a natural number j such that $1 \leqslant j$ and $j \leqslant k+1$ and $g(i) = f(j)$,

where a_4 = intpos 4, a_6 = intpos 6, $I_2 = W_1$, W_1 = while $>$ 0(GBP, 6, B_1), $B_1 = k_1$; k_2; I_1, k_1 = (GBP, 5) := (intpos 4, -1), k_2 = SubFrom(GBP, 5, intpos 4, 0), I_1 = **if** GBP $>$ 5 **then** T_1 **else** F_1, $T_1 = k_3$; k_4; k_5; k_6; k_7, k_3 = (GBP, 5) := (intpos 4, -1), k_4 = (intpos 4, -1) := (intpos 4, 0), k_5 = (intpos 4, 0) := (GBP, 5), k_6 = AddTo(GBP, 4, -1), k_7 = AddTo(GBP, 6, -1), and F_1 = Load((GBP)$_6$:=0).

(18) Let s be a state of SCMPDS, f, g be finite sequences of elements of $\mathbb{Z}$, and p_0, n be natural numbers. Suppose $p_0 \geqslant 6$ and len $f = n$ and len $g = n$ and f is FinSequence on s, p_0 and g is FinSequence on IExec(insert-sort$(n, p_0 + 1), s)$, p_0. Then f and g are fiberwise equipotent and g is non decreasing on 1, n.

References

[1] Grzegorz Bancerek. Cardinal numbers. *Formalized Mathematics*, 1(**2**):377–382, 1990.

[2] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[3] Grzegorz Bancerek. König's theorem. *Formalized Mathematics*, 1(**3**):589–593, 1990.

[4] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[5] Jing-Chao Chen. Computation and program shift in the SCMPDS computer. *Formalized Mathematics*, 8(**1**):193–199, 1999.

[6] Jing-Chao Chen. Computation of two consecutive program blocks for SCMPDS. *Formalized Mathematics*, 8(**1**):211–217, 1999.

[7] Jing-Chao Chen. The construction and computation of conditional statements for SCMPDS. *Formalized Mathematics*, 8(**1**):219–234, 1999.

[8] Jing-Chao Chen. The construction and shiftability of program blocks for SCMPDS. *Formalized Mathematics*, 8(**1**):201–210, 1999.

[9] Jing-Chao Chen. The SCMPDS computer and the basic semantics of its instructions. *Formalized Mathematics*, 8(**1**):183–191, 1999.

[10] Jing-Chao Chen. The construction and computation of while-loop programs for SCMPDS. *Formalized Mathematics*, 9(**2**):397–405, 2001.

[11] Jing-Chao Chen. Recursive Euclide algorithm. *Formalized Mathematics*, 9(**1**):1–4, 2001.

[12] Jing-Chao Chen and Piotr Rudnicki. The construction and computation of for-loop programs for SCMPDS. *Formalized Mathematics*, 9(**1**):209–219, 2001.

[13] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[14] Andrzej Kondracki. The Chinese Remainder Theorem. *Formalized Mathematics*, 6(**4**):573–577, 1997.

[15] Jarosław Kotowicz. Functions and finite sequences of real numbers. *Formalized Mathematics*, 3(**2**):275–278, 1992.

[16] Yatsuka Nakamura and Andrzej Trybulec. A mathematical model of CPU. *Formalized Mathematics*, 3(**2**):151–160, 1992.

[17] Piotr Rudnicki. The `for` (going up) macro instruction. *Formalized Mathematics*, 7(**1**):107–114, 1998.

[18] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[19] Andrzej Trybulec and Yatsuka Nakamura. Some remarks on the simple concrete model of computer. *Formalized Mathematics*, 4(**1**):51–56, 1993.

[20] Michał J. Trybulec. Integers. *Formalized Mathematics*, 1(**3**):501–505, 1990.

[21] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

Received June 14, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Quick Sort on SCMPDS[1]

Jing-Chao Chen
Shanghai Jiaotong University / China Bell Labs

Summary. Proving the correctness of quick sort is much more complicated than proving the correctness of the insert sort. Its difficulty is determined mainly by the following points:

- Quick sort needs to use a push-down stack.
- It contains three nested loops.
- A subroutine of this algorithm, "Partition", has no loop-invariant.

This means we cannot justify the correctness of the "Partition" subroutine by the Hoare's axiom on program verification. This article is organized as follows. First, we present several fundamental properties of "while" program and finite sequence. Second, we define the "Partition" subroutine on SCMPDS, the task of which is to split a sequence into a smaller and a larger subsequence. The definition of quick sort on SCMPDS follows. Finally, we describe the basic property of the "Partition" and quick sort, and prove their correctness.

MML Identifier: SCPQSORT.

The terminology and notation used here have been introduced in the following articles: [18], [19], [23], [21], [1], [3], [4], [6], [24], [2], [15], [26], [17], [11], [7], [10], [8], [9], [12], [14], [5], [13], [20], [25], [22], and [16].

1. The Several Properties of "while" Program and Finite Sequence

In this paper n, p_0 denote natural numbers.

Let I, J be shiftable Program-blocks, let a be an Int position, and let k_1 be an integer. Observe that **if** $a > k_1$ **then** I **else** J is shiftable.

Next we state the proposition

[1] This research is partially supported by the National Natural Science Foundation of China Grant No. 69873033.

(1) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, J be a shiftable Program-block, a, b be Int positions, and k_1 be an integer. Suppose $s(\mathrm{DataLoc}(s(a), k_1)) > 0$ and I is closed on s and halting on s. Then $(\mathrm{IExec}(\textbf{if } a > k_1 \textbf{ then } I \textbf{ else } J, s))(b) = (\mathrm{IExec}(I, s))(b)$.

One can prove the following propositions:

(2) Let s, s_1 be states of SCMPDS, I be a No-StopCode shiftable Program-block, a be an Int position, i be an integer, and m be a natural number. Suppose $\mathrm{card}\, I > 0$ and I is closed on s and halting on s and $s(\mathrm{DataLoc}(s(a), i)) > 0$ and $m = \mathrm{LifeSpan}(s +\cdot \mathrm{Initialized}(\mathrm{stop}\, I)) + 2$ and $s_1 = (\mathrm{Computation}(s +\cdot \mathrm{Initialized}(\mathrm{stop\, while} > 0(a, i, I))))(m)$. Then $s_1 \restriction \text{Data-Loc}_{\mathrm{SCM}} = \mathrm{IExec}(I, s) \restriction \text{Data-Loc}_{\mathrm{SCM}}$ and $s_1 +\cdot \mathrm{Initialized}(\mathrm{stop\, while} > 0(a, i, I)) = s_1$.

(3) Let s be a state of SCMPDS and I be a Program-block. Suppose that for every state t of SCMPDS such that $t \restriction \text{Data-Loc}_{\mathrm{SCM}} = s \restriction \text{Data-Loc}_{\mathrm{SCM}}$ holds I is halting on t. Then I is closed on s.

(4) For all instructions i_1, i_2, i_3, i_4 of SCMPDS holds $\mathrm{card}(i_1;\, i_2;\, i_3;\, i_4) = 4$.

(5) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a, x, y be Int positions, and i, c be integers. Suppose that

(i) $\mathrm{card}\, I > 0$,

(ii) $s(x) \geqslant c + s(\mathrm{DataLoc}(s(a), i))$, and

(iii) for every state t of SCMPDS such that $t(x) \geqslant c + t(\mathrm{DataLoc}(s(a), i))$ and $t(y) = s(y)$ and $t(a) = s(a)$ and $t(\mathrm{DataLoc}(s(a), i)) > 0$ holds $(\mathrm{IExec}(I, t))(a) = t(a)$ and I is closed on t and halting on t and $(\mathrm{IExec}(I, t))(\mathrm{DataLoc}(s(a), i)) < t(\mathrm{DataLoc}(s(a), i))$ and $(\mathrm{IExec}(I, t))(x) \geqslant c + (\mathrm{IExec}(I, t))(\mathrm{DataLoc}(s(a), i))$ and $(\mathrm{IExec}(I, t))(y) = t(y)$.

Then while $> 0(a, i, I)$ is closed on s and while $> 0(a, i, I)$ is halting on s and if $s(\mathrm{DataLoc}(s(a), i)) > 0$, then $\mathrm{IExec}(\mathrm{while} > 0(a, i, I), s) = \mathrm{IExec}(\mathrm{while} > 0(a, i, I), \mathrm{IExec}(I, s))$.

(6) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a, x, y be Int positions, and i, c be integers. Suppose that

(i) $\mathrm{card}\, I > 0$,

(ii) $s(x) \geqslant c$, and

(iii) for every state t of SCMPDS such that $t(x) \geqslant c$ and $t(y) = s(y)$ and $t(a) = s(a)$ and $t(\mathrm{DataLoc}(s(a), i)) > 0$ holds $(\mathrm{IExec}(I, t))(a) = t(a)$ and I is closed on t and halting on t and $(\mathrm{IExec}(I, t))(\mathrm{DataLoc}(s(a), i)) < t(\mathrm{DataLoc}(s(a), i))$ and $(\mathrm{IExec}(I, t))(x) \geqslant c$ and $(\mathrm{IExec}(I, t))(y) = t(y)$.

Then while $> 0(a, i, I)$ is closed on s and while $> 0(a, i, I)$ is halting on s and if $s(\mathrm{DataLoc}(s(a), i)) > 0$, then $\mathrm{IExec}(\mathrm{while} > 0(a, i, I), s) = \mathrm{IExec}(\mathrm{while} > 0(a, i, I), \mathrm{IExec}(I, s))$.

(7) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a, x_1, x_2, x_3, x_4 be Int positions, and i, c, m_1 be integers. Suppose that

(i) $\mathrm{card}\, I > 0$,

(ii) $s(x_4) = (s(x_3) - c) + s(x_1)$,

(iii) $m_1 \leqslant s(x_3) - c$, and

(iv) for every state t of SCMPDS such that $t(x_4) = (t(x_3) - c) + t(x_1)$ and $m_1 \leqslant t(x_3) - c$ and $t(x_2) = s(x_2)$ and $t(a) = s(a)$ and $t(\mathrm{DataLoc}(s(a), i)) > 0$ holds $(\mathrm{IExec}(I, t))(a) = t(a)$ and I is closed on t and halting on t and $(\mathrm{IExec}(I, t))(\mathrm{DataLoc}(s(a), i)) < t(\mathrm{DataLoc}(s(a), i))$ and $(\mathrm{IExec}(I, t))(x_4) = ((\mathrm{IExec}(I, t))(x_3) - c) + (\mathrm{IExec}(I, t))(x_1)$ and $m_1 \leqslant (\mathrm{IExec}(I, t))(x_3) - c$ and $(\mathrm{IExec}(I, t))(x_2) = t(x_2)$.

Then while $> 0(a, i, I)$ is closed on s and while $> 0(a, i, I)$ is halting on s and if $s(\mathrm{DataLoc}(s(a), i)) > 0$, then $\mathrm{IExec}(\text{while} > 0(a, i, I), s) = \mathrm{IExec}(\text{while} > 0(a, i, I), \mathrm{IExec}(I, s))$.

(8) Let f be a finite sequence of elements of $\mathbb{Z}$ and m, k_1, k, n be natural numbers. Suppose that $m \leqslant k$ and $k \leqslant n$ and $k_1 = k - 1$ and f is non decreasing on m, k_1 and f is non decreasing on $k + 1$, n and for every natural number i such that $m \leqslant i$ and $i < k$ holds $f(i) \leqslant f(k)$ and for every natural number i such that $k < i$ and $i \leqslant n$ holds $f(k) \leqslant f(i)$. Then f is non decreasing on m, n.

(9) Let f, g be finite sequences and x be arbitrary. Suppose $x \in \mathrm{dom}\, g$ and f and g are fiberwise equipotent. Then there exists arbitrary y such that $y \in \mathrm{dom}\, g$ and $f(x) = g(y)$.

(10) Let f, g, h be finite sequences. Then f and g are fiberwise equipotent if and only if $h \frown f$ and $h \frown g$ are fiberwise equipotent.

(11) Let f, g be finite sequences and m, n, j be natural numbers. Suppose that f and g are fiberwise equipotent and $m \leqslant n$ and $n \leqslant \mathrm{len}\, f$ and for every natural number i such that $1 \leqslant i$ and $i \leqslant m$ holds $f(i) = g(i)$ and for every natural number i such that $n < i$ and $i \leqslant \mathrm{len}\, f$ holds $f(i) = g(i)$ and $m < j$ and $j \leqslant n$. Then there exists a natural number k such that $m < k$ and $k \leqslant n$ and $f(j) = g(k)$.

2. Program Partition is to Split a Sequence into a Smaller and a Larger Subsequence

The Program-block Partition is defined by the condition (Def. 1).

(Def. 1) Partition = ((GBP, 5) := (GBP, 4)); SubFrom(GBP, 5, GBP, 2); ((GBP, 3) := (GBP, 2)); AddTo(GBP, 3, 1); while $>$ 0(GBP, 5, while $>$ 0(GBP, 5, ((GBP, 7) := (GBP, 5)); AddTo(GBP, 5, -1); ((GBP, 6) := (intpos 4, 0)); SubFrom(GBP, 6, intpos 2, 0); (**if** GBP $>$ 6 **then**

AddTo(GBP, 4, −1); AddTo(GBP, 7, −1) **else** Load($(\mathrm{GBP})_{5}$:=0))); while > 0(GBP, 7, ((GBP, 5) := (GBP, 7)); AddTo(GBP, 7, −1); ((GBP, 6) := (intpos 2, 0)); SubFrom(GBP, 6, intpos 3, 0); (**if** GBP > 6 **then** AddTo(GBP, 3, 1); AddTo(GBP, 5, −1) **else** Load($(\mathrm{GBP})_{7}$:=0))); (**if** GBP > 0 **then** 5 **else** (((GBP, 6) := (intpos 4, 0)); ((intpos 4, 0) := (intpos 3, 0)); ((intpos 3, 0) := (GBP, 6)); AddTo(GBP, 5, −2); AddTo(GBP, 3, 1); AddTo(GBP, 4, −1)))); ((GBP, 6) := (intpos 4, 0)); ((intpos 4, 0) := (intpos 2, 0)); ((intpos 2, 0) := (GBP, 6)).

3. The Construction of Quick Sort

Let n, p_0 be natural numbers. The functor QuickSort(n, p_0) yielding a Program-block is defined by the condition (Def. 2).

(Def. 2) QuickSort(n, p_0) = (GBP :=0); (SBP :=1); ($(\mathrm{SBP})_{p_1}$:=$p_0 + 1$); ($(\mathrm{SBP})_{p_1+1}$:=p_1); while > 0(GBP, 1, ((GBP, 2) := (SBP, $p_1 + 1$)); SubFrom(GBP, 2, SBP, p_1); (**if** GBP > 2 **then** ((GBP, 2) := (SBP, p_1)); ((GBP, 4) := (SBP, $p_1 + 1$)); Partition ; (((SBP, $p_1 + 3$) := (SBP, $p_1 + 1$)); ((SBP, $p_1 + 1$) := (GBP, 4)); ((SBP, $p_1 + 2$) := (GBP, 4)); AddTo(SBP, $p_1 + 1$, −1); AddTo(SBP, $p_1 + 2$, 1); AddTo(GBP, 1, 2)) **else** Load(AddTo(GBP, 1, −2)))), where $p_1 = p_0 + n$.

4. The Basic Property of Partition Program

The following four propositions are true:

(12) card Partition = 38.

(13) Let s be a state of SCMPDS and m_1, p_0 be natural numbers. Suppose $s(\mathrm{GBP}) = 0$ and $s(\mathrm{intpos}\,4) - s(\mathrm{intpos}\,2) > 0$ and $s(\mathrm{intpos}\,2) = m_1$ and $m_1 \geqslant p_0 + 1$ and $p_0 \geqslant 7$. Then Partition is closed on s and Partition is halting on s.

(14) Let s be a state of SCMPDS, m_1, p_0, n be natural numbers, and f, f_1 be finite sequences of elements of $\mathbb{Z}$. Suppose that $s(\mathrm{GBP}) = 0$ and $s(\mathrm{intpos}\,4) - s(\mathrm{intpos}\,2) > 0$ and $s(\mathrm{intpos}\,2) = m_1$ and $m_1 \geqslant p_0 + 1$ and $s(\mathrm{intpos}\,4) \leqslant p_0 + n$ and $p_0 \geqslant 7$ and f is FinSequence on s, p_0 and len $f = n$ and f_1 is FinSequence on IExec(Partition, s), p_0 and len $f_1 = n$. Then

(i) (IExec(Partition, s))(GBP) = 0,

(ii) (IExec(Partition, s))(intpos 1) = s(intpos 1),

(iii) f and f_1 are fiberwise equipotent, and

(iv) there exists a natural number m_4 such that m_4 = (IExec(Partition, s))

(intpos 4) and $m_1 \leqslant m_4$ and $m_4 \leqslant s(\mathrm{intpos}\,4)$ and for every natural number i such that $m_1 \leqslant i$ and $i < m_4$ holds $(\mathrm{IExec}(\mathrm{Partition}, s))(\mathrm{intpos}\, m_4) \geqslant (\mathrm{IExec}(\mathrm{Partition}, s))(\mathrm{intpos}\, i)$ and for every natural number i such that $m_4 < i$ and $i \leqslant s(\mathrm{intpos}\,4)$ holds $(\mathrm{IExec}(\mathrm{Partition}, s))(\mathrm{intpos}\, m_4) \leqslant (\mathrm{IExec}(\mathrm{Partition}, s))(\mathrm{intpos}\, i)$ and for every natural number i such that $i \geqslant p_0 + 1$ but $i < s(\mathrm{intpos}\,2)$ or $i > s(\mathrm{intpos}\,4)$ holds $(\mathrm{IExec}(\mathrm{Partition}, s))(\mathrm{intpos}\, i) = s(\mathrm{intpos}\, i)$.

(15) Partition is No-StopCode and shiftable.

5. The Basic Property of Quick Sort and Its Correctness

One can prove the following three propositions:

(16) card QuickSort$(n, p_0) = 57$.

(17) For all natural numbers p_0, n such that $p_0 \geqslant 7$ holds QuickSort(n, p_0) is parahalting.

(18) Let s be a state of SCMPDS and p_0, n be natural numbers. Suppose $p_0 \geqslant 7$. Then there exist finite sequences f, g of elements of $\mathbb{Z}$ such that

(i) $\mathrm{len}\, f = n$,

(ii) f is FinSequence on s, p_0,

(iii) $\mathrm{len}\, g = n$,

(iv) g is FinSequence on IExec(QuickSort$(n, p_0), s)$, p_0,

(v) f and g are fiberwise equipotent, and

(vi) g is non decreasing on 1, n.

References

[1] Grzegorz Bancerek. Cardinal numbers. *Formalized Mathematics*, 1(**2**):377–382, 1990.

[2] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[3] Grzegorz Bancerek. König's theorem. *Formalized Mathematics*, 1(**3**):589–593, 1990.

[4] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[5] Grzegorz Bancerek and Piotr Rudnicki. Development of terminology for **scm**. *Formalized Mathematics*, 4(**1**):61–67, 1993.

[6] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[7] Jing-Chao Chen. Computation and program shift in the SCMPDS computer. *Formalized Mathematics*, 8(**1**):193–199, 1999.

[8] Jing-Chao Chen. Computation of two consecutive program blocks for SCMPDS. *Formalized Mathematics*, 8(**1**):211–217, 1999.

[9] Jing-Chao Chen. The construction and computation of conditional statements for SCMPDS. *Formalized Mathematics*, 8(**1**):219–234, 1999.

[10] Jing-Chao Chen. The construction and shiftability of program blocks for SCMPDS. *Formalized Mathematics*, 8(**1**):201–210, 1999.

[11] Jing-Chao Chen. The SCMPDS computer and the basic semantics of its instructions. *Formalized Mathematics*, 8(**1**):183–191, 1999.

[12] Jing-Chao Chen. The construction and computation of while-loop programs for SCMPDS. *Formalized Mathematics*, 9(**2**):397–405, 2001.

[13] Jing-Chao Chen. Insert sort on SCMPDS. *Formalized Mathematics*, 9(**2**):407–412, 2001.

[14] Jing-Chao Chen. Recursive Euclide algorithm. *Formalized Mathematics*, 9(**1**):1–4, 2001.

[15] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[16] Andrzej Kondracki. The Chinese Remainder Theorem. *Formalized Mathematics*, 6(**4**):573–577, 1997.

[17] Jarosław Kotowicz. Functions and finite sequences of real numbers. *Formalized Mathematics*, 3(**2**):275–278, 1992.

[18] Yatsuka Nakamura and Andrzej Trybulec. A mathematical model of CPU. *Formalized Mathematics*, 3(**2**):151–160, 1992.

[19] Yatsuka Nakamura and Andrzej Trybulec. On a mathematical model of programs. *Formalized Mathematics*, 3(**2**):241–250, 1992.

[20] Piotr Rudnicki. The `for` (going up) macro instruction. *Formalized Mathematics*, 7(**1**):107–114, 1998.

[21] Yasushi Tanaka. On the decomposition of the states of SCM. *Formalized Mathematics*, 5(**1**):1–8, 1996.

[22] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[23] Andrzej Trybulec and Yatsuka Nakamura. Some remarks on the simple concrete model of computer. *Formalized Mathematics*, 4(**1**):51–56, 1993.

[24] Michał J. Trybulec. Integers. *Formalized Mathematics*, 1(**3**):501–505, 1990.

[25] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[26] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

Received June 14, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

Justifying the Correctness of the Fibonacci Sequence and the Euclide Algorithm by Loop-Invariant[1]

Jing-Chao Chen
Shanghai Jiaotong University

Summary. If a loop-invariant exists in a loop program, computing its result by loop-invariant is simpler and easier than computing its result by the inductive method. For this purpose, the article describes the premise and the final computation result of the program such as "while<0", "while>0", "while<>0" by loop-invariant. To test the effectiveness of the computation method given in this article, by using loop-invariant of the loop programs mentioned above, we justify the correctness of the following three examples: Summing n integers (used for testing "while>0"), Fibonacci sequence (used for testing "while<0"), Greatest Common Divisor, i.e. Euclide algorithm (used for testing "while<>0").

MML Identifier: SCPINVAR.

The notation and terminology used here have been introduced in the following papers: [18], [22], [19], [1], [3], [4], [6], [7], [24], [23], [2], [5], [16], [26], [27], [12], [8], [11], [9], [10], [13], [15], [14], [21], [25], [20], and [17].

1. Preliminaries

For simplicity, we adopt the following rules: m, n are natural numbers, i, j are instructions of SCMPDS, I is a Program-block, and a is an Int position.

One can prove the following propositions:

(1) For all natural numbers n, m, l such that $n \mid m$ and $n \mid l$ holds $n \mid m - l$.

[1] This research is partially supported by the National Natural Science Foundation of China Grant No. 69873033.

(2) $m \mid n$ iff $m \mid n$ **qua** integer.

(3) $\gcd(m, n) = \gcd(m, |n - m|)$.

(4) For all integers a, b such that $a \geqslant 0$ and $b \geqslant 0$ holds $a \gcd b = a \gcd b - a$.

(5) $(i;\, j;\, I)(\mathrm{inspos}\, 0) = i$ and $(i;\, j;\, I)(\mathrm{inspos}\, 1) = j$.

(6) Let a, b be Int positions. Then there exists a function f from $\prod$(the object kind of SCMPDS) into $\mathbb{N}$ such that for every state s of SCMPDS holds

(i) if $s(a) = s(b)$, then $f(s) = 0$, and

(ii) if $s(a) \neq s(b)$, then $f(s) = \max(|s(a)|, |s(b)|)$.

(7) There exists a function f from $\prod$(the object kind of SCMPDS) into $\mathbb{N}$ such that for every state s of SCMPDS holds

(i) if $s(a) \geqslant 0$, then $f(s) = 0$, and

(ii) if $s(a) < 0$, then $f(s) = -s(a)$.

2. Computing Directly the Result of "while<0" Program by Loop-Invariant

The scheme *WhileLEnd* deals with a unary functor $\mathcal{F}$ yielding a natural number, a state $\mathcal{A}$ of SCMPDS, a No-StopCode shiftable Program-block $\mathcal{B}$, an Int position $\mathcal{C}$, an integer $\mathcal{D}$, and a unary predicate $\mathcal{P}$, and states that:

$\mathcal{F}(\mathcal{A}) = \mathcal{F}(\mathcal{A})$ or $\mathcal{P}[\mathcal{A}]$ but $\mathcal{F}(\mathrm{Dstate\, IExec}(\mathrm{while} < 0(\mathcal{C}, \mathcal{D}, \mathcal{B}), \mathcal{A})) = 0$ but $\mathcal{P}[\mathrm{Dstate\, IExec}(\mathrm{while} < 0(\mathcal{C}, \mathcal{D}, \mathcal{B}), \mathcal{A})]$

provided the parameters satisfy the following conditions:

- $\mathrm{card}\, \mathcal{B} > 0$,
- For every state t of SCMPDS such that $\mathcal{P}[\mathrm{Dstate}\, t]$ holds $\mathcal{F}(\mathrm{Dstate}\, t) = 0$ iff $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) \geqslant 0$,
- $\mathcal{P}[\mathrm{Dstate}\, \mathcal{A}]$, and
- Let t be a state of SCMPDS. Suppose $\mathcal{P}[\mathrm{Dstate}\, t]$ and $t(\mathcal{C}) = \mathcal{A}(\mathcal{C})$ and $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) < 0$. Then $(\mathrm{IExec}(\mathcal{B}, t))(\mathcal{C}) = t(\mathcal{C})$ and $\mathcal{B}$ is closed on t and $\mathcal{B}$ is halting on t and $\mathcal{F}(\mathrm{Dstate\, IExec}(\mathcal{B}, t)) < \mathcal{F}(\mathrm{Dstate}\, t)$ and $\mathcal{P}[\mathrm{Dstate\, IExec}(\mathcal{B}, t)]$.

3. An Example: Summing Directly n Integers by Loop-Invariant

Let n, p_0 be natural numbers. The functor $\mathrm{sum}(n, p_0)$ yields a Program-block and is defined as follows:

(Def. 1) $\mathrm{sum}(n, p_0) = (\mathrm{GBP} := 0);\ (\mathrm{intpos}\, 1 := 0);\ (\mathrm{intpos}\, 2 := -n);\ (\mathrm{intpos}\, 3 := p_0 + 1);\ \mathrm{while} < 0(\mathrm{GBP}, 2, \mathrm{AddTo}(\mathrm{GBP}, 1, \mathrm{intpos}\, 3, 0);\ \mathrm{AddTo}(\mathrm{GBP}, 2, 1);\ \mathrm{AddTo}(\mathrm{GBP}, 3, 1))$.

We now state the proposition

(8) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a, b, c be Int positions, n, i, p_0 be natural numbers, and f be a finite sequence of elements of $\mathbb{Z}$. Suppose that $\operatorname{card} I > 0$ and f is FinSequence on s, p_0 and $\operatorname{len} f = n$ and $s(b) = 0$ and $s(a) = 0$ and $s(\operatorname{intpos} i) = -n$ and $s(c) = p_0 + 1$ and for every state t of SCMPDS such that there exists a finite sequence g of elements of $\mathbb{Z}$ such that g is FinSequence on s, p_0 and $\operatorname{len} g = t(\operatorname{intpos} i) + n$ and $t(b) = \sum g$ and $t(c) = p_0 + 1 + \operatorname{len} g$ and $t(a) = 0$ and $t(\operatorname{intpos} i) < 0$ and for every natural number i such that $i > p_0$ holds $t(\operatorname{intpos} i) = s(\operatorname{intpos} i)$ holds $(\operatorname{IExec}(I, t))(a) = 0$ and I is closed on t and halting on t and $(\operatorname{IExec}(I, t))(\operatorname{intpos} i) = t(\operatorname{intpos} i) + 1$ and there exists a finite sequence g of elements of $\mathbb{Z}$ such that g is FinSequence on s, p_0 and $\operatorname{len} g = t(\operatorname{intpos} i) + n + 1$ and $(\operatorname{IExec}(I, t))(c) = p_0 + 1 + \operatorname{len} g$ and $(\operatorname{IExec}(I, t))(b) = \sum g$ and for every natural number i such that $i > p_0$ holds $(\operatorname{IExec}(I, t))(\operatorname{intpos} i) = s(\operatorname{intpos} i)$. Then $(\operatorname{IExec}(\text{while} < 0(a, i, I), s))(b) = \sum f$ and while $< 0(a, i, I)$ is closed on s and while $< 0(a, i, I)$ is halting on s.

One can prove the following proposition

(9) Let s be a state of SCMPDS, n, p_0 be natural numbers, and f be a finite sequence of elements of $\mathbb{Z}$. Suppose $p_0 \geqslant 3$ and f is FinSequence on s, p_0 and $\operatorname{len} f = n$. Then $(\operatorname{IExec}(\operatorname{sum}(n, p_0), s))(\operatorname{intpos} 1) = \sum f$ and $\operatorname{sum}(n, p_0)$ is parahalting.

4. Computing Directly the Result of "while>0" Program by Loop-Invariant

The scheme *WhileGEnd* deals with a unary functor $\mathcal{F}$ yielding a natural number, a state $\mathcal{A}$ of SCMPDS, a No-StopCode shiftable Program-block $\mathcal{B}$, an Int position $\mathcal{C}$, an integer $\mathcal{D}$, and a unary predicate $\mathcal{P}$, and states that:

$\mathcal{F}(\mathcal{A}) = \mathcal{F}(\mathcal{A})$ or $\mathcal{P}[\mathcal{A}]$ but $\mathcal{F}(\operatorname{Dstate} \operatorname{IExec}(\text{while} > 0(\mathcal{C}, \mathcal{D}, \mathcal{B}), \mathcal{A})) = 0$ but $\mathcal{P}[\operatorname{Dstate} \operatorname{IExec}(\text{while} > 0(\mathcal{C}, \mathcal{D}, \mathcal{B}), \mathcal{A})]$

provided the parameters meet the following requirements:

- $\operatorname{card} \mathcal{B} > 0$,
- For every state t of SCMPDS such that $\mathcal{P}[\operatorname{Dstate} t]$ holds $\mathcal{F}(\operatorname{Dstate} t) = 0$ iff $t(\operatorname{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) \leqslant 0$,
- $\mathcal{P}[\operatorname{Dstate} \mathcal{A}]$, and
- Let t be a state of SCMPDS. Suppose $\mathcal{P}[\operatorname{Dstate} t]$ and $t(\mathcal{C}) = \mathcal{A}(\mathcal{C})$ and $t(\operatorname{DataLoc}(\mathcal{A}(\mathcal{C}), \mathcal{D})) > 0$. Then $(\operatorname{IExec}(\mathcal{B}, t))(\mathcal{C}) = t(\mathcal{C})$ and $\mathcal{B}$ is closed on t and $\mathcal{B}$ is halting on t and $\mathcal{F}(\operatorname{Dstate} \operatorname{IExec}(\mathcal{B}, t)) < \mathcal{F}(\operatorname{Dstate} t)$ and $\mathcal{P}[\operatorname{Dstate} \operatorname{IExec}(\mathcal{B}, t)]$.

5. An Example: Computing Directly Fibonacci Sequence by Loop-Invariant

Let n be a natural number. The functor Fib-macro n yields a Program-block and is defined by:

(Def. 2) Fib-macro n = (GBP :=0); (intpos 1:=0); (intpos 2:=1); (intpos 3:=n); while $> 0(\mathrm{GBP}, 3, ((\mathrm{GBP}, 4) := (\mathrm{GBP}, 2));$ $\mathrm{AddTo}(\mathrm{GBP}, 2, \mathrm{GBP}, 1);$ $((\mathrm{GBP}, 1) := (\mathrm{GBP}, 4));$ $\mathrm{AddTo}(\mathrm{GBP}, 3, -1))$.

We now state the proposition

(10) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a, f_0, f_1 be Int positions, and n, i be natural numbers. Suppose that

(i) card $I > 0$,

(ii) $s(a) = 0$,

(iii) $s(f_0) = 0$,

(iv) $s(f_1) = 1$,

(v) $s(\mathrm{intpos}\, i) = n$, and

(vi) for every state t of SCMPDS and for every natural number k such that $n = t(\mathrm{intpos}\, i) + k$ and $t(f_0) = \mathrm{Fib}(k)$ and $t(f_1) = \mathrm{Fib}(k+1)$ and $t(a) = 0$ and $t(\mathrm{intpos}\, i) > 0$ holds $(\mathrm{IExec}(I, t))(a) = 0$ and I is closed on t and halting on t and $(\mathrm{IExec}(I, t))(\mathrm{intpos}\, i) = t(\mathrm{intpos}\, i) - 1$ and $(\mathrm{IExec}(I, t))(f_0) = \mathrm{Fib}(k+1)$ and $(\mathrm{IExec}(I, t))(f_1) = \mathrm{Fib}(k+1+1)$.

Then $(\mathrm{IExec}(\text{while} > 0(a, i, I), s))(f_0) = \mathrm{Fib}(n)$ and $(\mathrm{IExec}(\text{while} > 0(a, i, I), s))(f_1) = \mathrm{Fib}(n+1)$ and while $> 0(a, i, I)$ is closed on s and while $> 0(a, i, I)$ is halting on s.

One can prove the following proposition

(11) For every state s of SCMPDS and for every natural number n holds $(\mathrm{IExec}(\text{Fib-macro}\, n, s))(\mathrm{intpos}\, 1) = \mathrm{Fib}(n)$ and $(\mathrm{IExec}(\text{Fib-macro}\, n, s))(\mathrm{intpos}\, 2) = \mathrm{Fib}(n+1)$ and Fib-macro n is parahalting.

6. The Construction of "while<>0" Loop Program

Let a be an Int position, let i be an integer, and let I be a Program-block. The functor while $<> 0(a, i, I)$ yields a Program-block and is defined as follows:

(Def. 3) while $<> 0(a, i, I) = ((a, i) <> 0\_\mathrm{goto}2)$; goto $(\mathrm{card}\, I + 2)$; I; goto $(-(\mathrm{card}\, I + 2))$.

7. The Basic Property of "while<>0" Program

One can prove the following propositions:

(12) For every Int position a and for every integer i and for every Program-block I holds $\mathrm{card}\,\mathrm{while} <> 0(a,i,I) = \mathrm{card}\,I + 3$.

(13) Let a be an Int position, i be an integer, m be a natural number, and I be a Program-block. Then $m < \mathrm{card}\,I + 3$ if and only if $\mathrm{inspos}\,m \in \mathrm{dom}\,\mathrm{while} <> 0(a,i,I)$.

(14) For every Int position a and for every integer i and for every Program-block I holds $\mathrm{inspos}\,0 \in \mathrm{dom}\,\mathrm{while} <> 0(a,i,I)$ and $\mathrm{inspos}\,1 \in \mathrm{dom}\,\mathrm{while} <> 0(a,i,I)$.

(15) Let a be an Int position, i be an integer, and I be a Program-block. Then $(\mathrm{while} <> 0(a,i,I))(\mathrm{inspos}\,0) = (a,i) <> 0\_\mathrm{goto}2$ and $(\mathrm{while} <> 0(a,i,I))(\mathrm{inspos}\,1) = \mathrm{goto}\,(\mathrm{card}\,I + 2)$ and $(\mathrm{while} <> 0(a,i,I))(\mathrm{inspos}\,\mathrm{card}\,I + 2) = \mathrm{goto}\,(-(\mathrm{card}\,I + 2))$.

(16) Let s be a state of SCMPDS, I be a Program-block, a be an Int position, and i be an integer. If $s(\mathrm{DataLoc}(s(a),i)) = 0$, then $\mathrm{while} <> 0(a,i,I)$ is closed on s and $\mathrm{while} <> 0(a,i,I)$ is halting on s.

(17) Let s be a state of SCMPDS, I be a Program-block, a, c be Int positions, and i be an integer. If $s(\mathrm{DataLoc}(s(a),i)) = 0$, then $\mathrm{IExec}(\mathrm{while} <> 0(a,i,I),s) = s+\cdot\,\mathrm{Start\text{-}At}(\mathrm{inspos}\,\mathrm{card}\,I + 3)$.

(18) Let s be a state of SCMPDS, I be a Program-block, a be an Int position, and i be an integer. If $s(\mathrm{DataLoc}(s(a),i)) = 0$, then $\mathbf{IC}_{\mathrm{IExec}(\mathrm{while}<>0(a,i,I),s)} = \mathrm{inspos}\,\mathrm{card}\,I + 3$.

(19) Let s be a state of SCMPDS, I be a Program-block, a, b be Int positions, and i be an integer. If $s(\mathrm{DataLoc}(s(a),i)) = 0$, then $(\mathrm{IExec}(\mathrm{while} <> 0(a,i,I),s))(b) = s(b)$.

Let I be a shiftable Program-block, let a be an Int position, and let i be an integer. Observe that $\mathrm{while} <> 0(a,i,I)$ is shiftable.

Let I be a No-StopCode Program-block, let a be an Int position, and let i be an integer. Note that $\mathrm{while} <> 0(a,i,I)$ is No-StopCode.

8. Computing Directly the Result of "while<>0" Program by Loop-Invariant

Now we present three schemes. The scheme *WhileNHalt* deals with a unary functor $\mathcal{F}$ yielding a natural number, a state $\mathcal{A}$ of SCMPDS, a No-StopCode shiftable Program-block $\mathcal{B}$, an Int position $\mathcal{C}$, an integer $\mathcal{D}$, and a unary predicate $\mathcal{P}$, and states that:

$\mathcal{F}(\mathcal{A}) = \mathcal{F}(\mathcal{A})$ or $\mathcal{P}[\mathcal{A}]$ but while $<> 0(\mathcal{C},\mathcal{D},\mathcal{B})$ is closed on $\mathcal{A}$ but while $<> 0(\mathcal{C},\mathcal{D},\mathcal{B})$ is halting on $\mathcal{A}$

provided the following conditions are satisfied:

- $\mathrm{card}\,\mathcal{B} > 0$,
- For every state t of SCMPDS such that $\mathcal{P}[\mathrm{Dstate}\,t]$ and $\mathcal{F}(\mathrm{Dstate}\,t) = 0$ holds $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}),\mathcal{D})) = 0$,
- $\mathcal{P}[\mathrm{Dstate}\,\mathcal{A}]$, and
- Let t be a state of SCMPDS. Suppose $\mathcal{P}[\mathrm{Dstate}\,t]$ and $t(\mathcal{C}) = \mathcal{A}(\mathcal{C})$ and $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}),\mathcal{D})) \neq 0$. Then $(\mathrm{IExec}(\mathcal{B},t))(\mathcal{C}) = t(\mathcal{C})$ and $\mathcal{B}$ is closed on t and $\mathcal{B}$ is halting on t and $\mathcal{F}(\mathrm{Dstate}\,\mathrm{IExec}(\mathcal{B},t)) < \mathcal{F}(\mathrm{Dstate}\,t)$ and $\mathcal{P}[\mathrm{Dstate}\,\mathrm{IExec}(\mathcal{B},t)]$.

The scheme *WhileNExec* deals with a unary functor $\mathcal{F}$ yielding a natural number, a state $\mathcal{A}$ of SCMPDS, a No-StopCode shiftable Program-block $\mathcal{B}$, an Int position $\mathcal{C}$, an integer $\mathcal{D}$, and a unary predicate $\mathcal{P}$, and states that:

$\mathcal{F}(\mathcal{A}) = \mathcal{F}(\mathcal{A})$ or $\mathcal{P}[\mathcal{A}]$ but $\mathrm{IExec}(\mathrm{while} <> 0(\mathcal{C},\mathcal{D},\mathcal{B}),\mathcal{A}) = \mathrm{IExec}(\mathrm{while} <> 0(\mathcal{C},\mathcal{D},\mathcal{B}),\mathrm{IExec}(\mathcal{B},\mathcal{A}))$

provided the parameters meet the following conditions:

- $\mathrm{card}\,\mathcal{B} > 0$,
- $\mathcal{A}(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}),\mathcal{D})) \neq 0$,
- For every state t of SCMPDS such that $\mathcal{P}[\mathrm{Dstate}\,t]$ and $\mathcal{F}(\mathrm{Dstate}\,t) = 0$ holds $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}),\mathcal{D})) = 0$,
- $\mathcal{P}[\mathrm{Dstate}\,\mathcal{A}]$, and
- Let t be a state of SCMPDS. Suppose $\mathcal{P}[\mathrm{Dstate}\,t]$ and $t(\mathcal{C}) = \mathcal{A}(\mathcal{C})$ and $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}),\mathcal{D})) \neq 0$. Then $(\mathrm{IExec}(\mathcal{B},t))(\mathcal{C}) = t(\mathcal{C})$ and $\mathcal{B}$ is closed on t and $\mathcal{B}$ is halting on t and $\mathcal{F}(\mathrm{Dstate}\,\mathrm{IExec}(\mathcal{B},t)) < \mathcal{F}(\mathrm{Dstate}\,t)$ and $\mathcal{P}[\mathrm{Dstate}\,\mathrm{IExec}(\mathcal{B},t)]$.

The scheme *WhileNEnd* deals with a unary functor $\mathcal{F}$ yielding a natural number, a state $\mathcal{A}$ of SCMPDS, a No-StopCode shiftable Program-block $\mathcal{B}$, an Int position $\mathcal{C}$, an integer $\mathcal{D}$, and a unary predicate $\mathcal{P}$, and states that:

$\mathcal{F}(\mathcal{A}) = \mathcal{F}(\mathcal{A})$ or $\mathcal{P}[\mathcal{A}]$ but $\mathcal{F}(\mathrm{Dstate}\,\mathrm{IExec}(\mathrm{while} <> 0(\mathcal{C},\mathcal{D},\mathcal{B}),\mathcal{A})) = 0$ but $\mathcal{P}[\mathrm{Dstate}\,\mathrm{IExec}(\mathrm{while} <> 0(\mathcal{C},\mathcal{D},\mathcal{B}),\mathcal{A})]$

provided the parameters satisfy the following conditions:

- $\mathrm{card}\,\mathcal{B} > 0$,
- For every state t of SCMPDS such that $\mathcal{P}[\mathrm{Dstate}\,t]$ holds $\mathcal{F}(\mathrm{Dstate}\,t) = 0$ iff $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}),\mathcal{D})) = 0$,
- $\mathcal{P}[\mathrm{Dstate}\,\mathcal{A}]$, and
- Let t be a state of SCMPDS. Suppose $\mathcal{P}[\mathrm{Dstate}\,t]$ and $t(\mathcal{C}) = \mathcal{A}(\mathcal{C})$ and $t(\mathrm{DataLoc}(\mathcal{A}(\mathcal{C}),\mathcal{D})) \neq 0$. Then $(\mathrm{IExec}(\mathcal{B},t))(\mathcal{C}) = t(\mathcal{C})$ and $\mathcal{B}$ is closed on t and $\mathcal{B}$ is halting on t and $\mathcal{F}(\mathrm{Dstate}\,\mathrm{IExec}(\mathcal{B},t)) < \mathcal{F}(\mathrm{Dstate}\,t)$ and $\mathcal{P}[\mathrm{Dstate}\,\mathrm{IExec}(\mathcal{B},t)]$.

We now state the proposition

(20) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a, b, c be Int positions, and i, d be integers. Suppose that

(i) $\operatorname{card} I > 0$,

(ii) $s(a) = d$,

(iii) $s(b) > 0$,

(iv) $s(c) > 0$,

(v) $s(\mathrm{DataLoc}(d, i)) = s(b) - s(c)$, and

(vi) for every state t of SCMPDS such that $t(b) > 0$ and $t(c) > 0$ and $t(a) = d$ and $t(\mathrm{DataLoc}(d, i)) = t(b) - t(c)$ and $t(b) \neq t(c)$ holds $(\mathrm{IExec}(I, t))(a) = d$ and I is closed on t and halting on t and if $t(b) > t(c)$, then $(\mathrm{IExec}(I, t))(b) = t(b) - t(c)$ and $(\mathrm{IExec}(I, t))(c) = t(c)$ and if $t(b) \leqslant t(c)$, then $(\mathrm{IExec}(I, t))(c) = t(c) - t(b)$ and $(\mathrm{IExec}(I, t))(b) = t(b)$ and $(\mathrm{IExec}(I, t))(\mathrm{DataLoc}(d, i)) = (\mathrm{IExec}(I, t))(b) - (\mathrm{IExec}(I, t))(c)$.

Then while <> $0(a, i, I)$ is closed on s and while <> $0(a, i, I)$ is halting on s and if $s(\mathrm{DataLoc}(s(a), i)) \neq 0$, then IExec(while <> $0(a, i, I), s) =$ IExec(while <> $0(a, i, I), \mathrm{IExec}(I, s))$.

9. An Example: Computing Greatest Common Divisor (Euclide Algorithm) by Loop-Invariant

The Program-block GCD-Algorithm is defined by:

(Def. 4) GCD-Algorithm = (GBP :=0); ((GBP, 3) := (GBP, 1)); SubFrom(GBP, 3, GBP, 2); while <> 0(GBP, 3, (**if** GBP > 3 **then** Load(SubFrom(GBP, 1, GBP, 2)) **else** Load(SubFrom(GBP, 2, GBP, 1))); ((GBP, 3) := (GBP, 1)); SubFrom(GBP, 3, GBP, 2)).

Next we state the proposition

(21) Let s be a state of SCMPDS, I be a No-StopCode shiftable Program-block, a, b, c be Int positions, and i, d be integers. Suppose that

(i) $\operatorname{card} I > 0$,

(ii) $s(a) = d$,

(iii) $s(b) > 0$,

(iv) $s(c) > 0$,

(v) $s(\mathrm{DataLoc}(d, i)) = s(b) - s(c)$, and

(vi) for every state t of SCMPDS such that $t(b) > 0$ and $t(c) > 0$ and $t(a) = d$ and $t(\mathrm{DataLoc}(d, i)) = t(b) - t(c)$ and $t(b) \neq t(c)$ holds $(\mathrm{IExec}(I, t))(a) = d$ and I is closed on t and halting on t and if $t(b) > t(c)$, then $(\mathrm{IExec}(I, t))(b) = t(b) - t(c)$ and $(\mathrm{IExec}(I, t))(c) = t(c)$ and if $t(b) \leqslant t(c)$, then $(\mathrm{IExec}(I, t))(c) = t(c) - t(b)$ and $(\mathrm{IExec}(I, t))(b) = t(b)$ and $(\mathrm{IExec}(I, t))(\mathrm{DataLoc}(d, i)) = (\mathrm{IExec}(I, t))(b) - (\mathrm{IExec}(I, t))(c)$.

Then (IExec(while $<>$ $0(a,i,I),s))(b) = s(b)\,\mathrm{gcd}\,s(c)$ and (IExec(while $<>$ $0(a,i,I),s))(c) = s(b)\,\mathrm{gcd}\,s(c)$.

We now state the proposition

(22) card GCD-Algorithm = 12.

The following proposition is true

(23) Let s be a state of SCMPDS and x, y be integers. Suppose s(intpos 1) $= x$ and s(intpos 2) $= y$ and $x > 0$ and $y > 0$. Then (IExec(GCD-Algorithm, s))(intpos 1) $= x\,\mathrm{gcd}\,y$ and (IExec(GCD-Algorithm, s))(intpos 2) $= x\,\mathrm{gcd}\,y$ and GCD-Algorithm is closed on s and GCD-Algorithm is halting on s.

References

[1] Grzegorz Bancerek. Cardinal numbers. *Formalized Mathematics*, 1(**2**):377–382, 1990.

[2] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[3] Grzegorz Bancerek. König's theorem. *Formalized Mathematics*, 1(**3**):589–593, 1990.

[4] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[5] Grzegorz Bancerek and Piotr Rudnicki. Two programs for **scm**. Part I - preliminaries. *Formalized Mathematics*, 4(**1**):69–72, 1993.

[6] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[7] Czesław Byliński. Functions from a set to a set. *Formalized Mathematics*, 1(**1**):153–164, 1990.

[8] Jing-Chao Chen. Computation and program shift in the SCMPDS computer. *Formalized Mathematics*, 8(**1**):193–199, 1999.

[9] Jing-Chao Chen. Computation of two consecutive program blocks for SCMPDS. *Formalized Mathematics*, 8(**1**):211–217, 1999.

[10] Jing-Chao Chen. The construction and computation of conditional statements for SCMPDS. *Formalized Mathematics*, 8(**1**):219–234, 1999.

[11] Jing-Chao Chen. The construction and shiftability of program blocks for SCMPDS. *Formalized Mathematics*, 8(**1**):201–210, 1999.

[12] Jing-Chao Chen. The SCMPDS computer and the basic semantics of its instructions. *Formalized Mathematics*, 8(**1**):183–191, 1999.

[13] Jing-Chao Chen. The construction and computation of while-loop programs for SCMPDS. *Formalized Mathematics*, 9(**2**):397–405, 2001.

[14] Jing-Chao Chen. Insert sort on SCMPDS. *Formalized Mathematics*, 9(**2**):407–412, 2001.

[15] Jing-Chao Chen. Recursive Euclide algorithm. *Formalized Mathematics*, 9(**1**):1–4, 2001.

[16] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(**1**):35–40, 1990.

[17] Andrzej Kondracki. The Chinese Remainder Theorem. *Formalized Mathematics*, 6(**4**):573–577, 1997.

[18] Yatsuka Nakamura and Andrzej Trybulec. A mathematical model of CPU. *Formalized Mathematics*, 3(**2**):151–160, 1992.

[19] Yasushi Tanaka. On the decomposition of the states of SCM. *Formalized Mathematics*, 5(**1**):1–8, 1996.

[20] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[21] Andrzej Trybulec and Czesław Byliński. Some properties of real numbers. *Formalized Mathematics*, 1(**3**):445–449, 1990.

[22] Andrzej Trybulec and Yatsuka Nakamura. Some remarks on the simple concrete model of computer. *Formalized Mathematics*, 4(**1**):51–56, 1993.

[23] Michał J. Trybulec. Integers. *Formalized Mathematics*, 1(**3**):501–505, 1990.

[24] Wojciech A. Trybulec. Groups. *Formalized Mathematics*, 1(**5**):821–827, 1990.
[25] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.
[26] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.
[27] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

Received June 14, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

On the Isomorphism between Finite Chains

Marta Pruszyńska
University of Białystok

Marek Dudzicz
University of Białystok

MML Identifier: ORDERS_4.

The notation and terminology used here are introduced in the following papers: [11], [1], [4], [8], [9], [7], [10], [3], [5], [2], and [6].

A relational structure is said to be a chain if:

(Def. 1) It is a connected non empty poset or it is empty.

One can verify that every relational structure which is empty is also reflexive, transitive, and antisymmetric.

One can verify that every chain is reflexive, transitive, and antisymmetric.

Let us note that there exists a chain which is non empty.

One can check that every non empty chain is connected.

Let L be a 1-sorted structure. We say that L is countable if and only if:

(Def. 2) The carrier of L is countable.

Let us observe that there exists a chain which is finite and non empty.

Let us mention that there exists a chain which is countable.

Let A be a connected non empty relational structure. Observe that every non empty relational substructure of A which is full is also connected.

Let A be a finite relational structure. Observe that every relational substructure of A is finite.

We now state the proposition

(1) For all natural numbers n, m such that $n \leqslant m$ holds $\langle n, \subseteq \rangle$ is a full relational substructure of $\langle m, \subseteq \rangle$.

Let L be a relational structure and let A, B be sets. We say that A, B form upper lower partition of L if and only if:

(Def. 3) $A \cup B =$ the carrier of L and for all elements a, b of L such that $a \in A$ and $b \in B$ holds $a < b$.

Next we state four propositions:

ISSN 1426–2630

(2) Let L be a relational structure and A, B be sets. If A, B form upper lower partition of L, then $A \cap B = \emptyset$.

(3) Let L be an upper-bounded antisymmetric non empty relational structure. Then (the carrier of L) $\setminus \{\top_L\}$, $\{\top_L\}$ form upper lower partition of L.

(4) Let L_1, L_2 be relational structures and f be a map from L_1 into L_2. Suppose f is isomorphic. Then

(i) the carrier of $L_1 \neq \emptyset$ iff the carrier of $L_2 \neq \emptyset$,

(ii) the carrier of $L_2 \neq \emptyset$ or the carrier of $L_1 = \emptyset$, and

(iii) the carrier of $L_1 = \emptyset$ iff the carrier of $L_2 = \emptyset$.

(5) Let L_1, L_2 be antisymmetric relational structures and A_1, B_1 be subsets of L_1. Suppose A_1, B_1 form upper lower partition of L_1. Let A_2, B_2 be subsets of L_2. Suppose A_2, B_2 form upper lower partition of L_2. Let f be a map from $\mathrm{sub}(A_1)$ into $\mathrm{sub}(A_2)$. Suppose f is isomorphic. Let g be a map from $\mathrm{sub}(B_1)$ into $\mathrm{sub}(B_2)$. Suppose g is isomorphic. Then there exists a map h from L_1 into L_2 such that $h = f {+}{\cdot}\, g$ and h is isomorphic.

Let n be a natural number. Observe that $n + 1$ is non empty.

The following proposition is true

(6) Let A be a finite chain and n be a natural number. If $\overline{\overline{\text{the carrier of } A}} = n$, then A and $\langle n, \subseteq \rangle$ are isomorphic.

References

[1] Grzegorz Bancerek. Cardinal numbers. *Formalized Mathematics*, 1(**2**):377–382, 1990.

[2] Grzegorz Bancerek. Bounds in posets and relational substructures. *Formalized Mathematics*, 6(**1**):81–91, 1997.

[3] Grzegorz Bancerek. Directed sets, nets, ideals, filters, and maps. *Formalized Mathematics*, 6(**1**):93–107, 1997.

[4] Czesław Byliński. The modification of a function by a function and the iteration of the composition of a function. *Formalized Mathematics*, 1(**3**):521–527, 1990.

[5] Czesław Byliński. Galois connections. *Formalized Mathematics*, 6(**1**):131–143, 1997.

[6] Adam Grabowski and Robert Milewski. Boolean posets, posets under inclusion and products of relational structures. *Formalized Mathematics*, 6(**1**):117–121, 1997.

[7] Beata Padlewska and Agata Darmochwał. Topological spaces and continuous functions. *Formalized Mathematics*, 1(**1**):223–230, 1990.

[8] Wojciech A. Trybulec. Groups. *Formalized Mathematics*, 1(**5**):821–827, 1990.

[9] Wojciech A. Trybulec. Partially ordered sets. *Formalized Mathematics*, 1(**2**):313–319, 1990.

[10] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[11] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

Received June 29, 2000

FORMALIZED MATHEMATICS
Volume 9, Number 2, 2001
University of Białystok

The Jónsson Theorem about the Representation of Modular Lattices

Mariusz Łapiński
University of Białystok

Summary. Formalization of [14, pp. 192–199], chapter IV. Partition Lattices, theorem 8.

MML Identifier: `LATTICE8`.

The articles [8], [18], [6], [9], [10], [3], [15], [20], [1], [21], [13], [2], [17], [7], [23], [24], [22], [19], [5], [12], [16], [4], [25], and [11] provide the terminology and notation for this paper.

1. Preliminaries

Let A be a non empty set and let P, R be binary relations on A. Let us observe that $P \subseteq R$ if and only if:

(Def. 1) For all elements a, b of A such that $\langle a, b\rangle \in P$ holds $\langle a, b\rangle \in R$.

Let L be a relational structure. We say that L is finitely typed if and only if the condition (Def. 2) is satisfied.

(Def. 2) There exists a non empty set A such that

(i) for every set e such that $e \in$ the carrier of L holds e is an equivalence relation of A, and

(ii) there exists a natural number o such that for all equivalence relations e_1, e_2 of A and for all sets x, y such that $e_1 \in$ the carrier of L and $e_2 \in$ the carrier of L and $\langle x, y\rangle \in e_1 \sqcup e_2$ there exists a non empty finite sequence F of elements of A such that $\operatorname{len} F = o$ and x and y are joint by F, e_1 and e_2.

ISSN 1426–2630

Let L be a lower-bounded lattice and let n be a natural number. We say that L has a representation of type $\leqslant n$ if and only if the condition (Def. 3) is satisfied.

(Def. 3) There exists a non trivial set A and there exists a homomorphism f from L to $\mathrm{EqRelPoset}(A)$ such that

(i) f is one-to-one,

(ii) $\mathrm{Im}\, f$ is finitely typed,

(iii) there exists an equivalence relation e of A such that $e \in$ the carrier of $\mathrm{Im}\, f$ and $e \neq \mathrm{id}_A$, and

(iv) the type of $\mathrm{Im}\, f \leqslant n$.

Let us mention that there exists a lattice which is lower-bounded, distributive, and finite.

Let A be a non trivial set. Observe that there exists a non empty sublattice of $\mathrm{EqRelPoset}(A)$ which is non trivial, finitely typed, and full.

One can prove the following propositions:

(1) For every non empty set A and for every lower-bounded lattice L and for every distance function d of A, L holds $\mathrm{succ}\,\emptyset \subseteq \mathrm{DistEsti}(d)$.

(2) Every trivial semilattice is modular.

(3) Let A be a non empty set and L be a non empty sublattice of $\mathrm{EqRelPoset}(A)$. Then L is trivial or there exists an equivalence relation e of A such that $e \in$ the carrier of L and $e \neq \mathrm{id}_A$.

(4) Let L_1, L_2 be lower-bounded lattices and f be a map from L_1 into L_2. Suppose f is infs-preserving and sups-preserving. Then f is meet-preserving and join-preserving.

(5) For all lower-bounded lattices L_1, L_2 such that L_1 and L_2 are isomorphic and L_1 is modular holds L_2 is modular.

(6) Let S be a lower-bounded non empty poset, T be a non empty poset, and f be a monotone map from S into T. Then $\mathrm{Im}\, f$ is lower-bounded.

(7) Let L be a lower-bounded lattice, x, y be elements of L, A be a non empty set, and f be a homomorphism from L to $\mathrm{EqRelPoset}(A)$. If f is one-to-one, then if $f^{\circ}(x) \leqslant f^{\circ}(y)$, then $x \leqslant y$.

2. The Jónsson Theorem

We now state two propositions:

(8) Let A be a non trivial set, L be a finitely typed full non empty sublattice of $\mathrm{EqRelPoset}(A)$, and e be an equivalence relation of A. Suppose $e \in$ the carrier of L and $e \neq \mathrm{id}_A$. If the type of $L \leqslant 2$, then L is modular.

(9) For every lower-bounded lattice L such that L has a representation of type $\leqslant 2$ holds L is modular.

Let A be a set. The functor new_set2 A is defined by:

(Def. 4) $\mathrm{new\_set2}\,A = A \cup \{\{A\}, \{\{A\}\}\}$.

Let A be a set. One can verify that new_set2 A is non empty.

Let A be a non empty set, let L be a lower-bounded lattice, let d be a bifunction from A into L, and let q be an element of $[: A, A,$ the carrier of $L,$ the carrier of $L :]$. The functor $\mathrm{new\_bi\_fun2}(d, q)$ yielding a bifunction from new_set2 A into L is defined by the conditions (Def. 5).

(Def. 5)(i) For all elements u, v of A holds $(\mathrm{new\_bi\_fun2}(d, q))(u, v) = d(u, v)$,

(ii) $(\mathrm{new\_bi\_fun2}(d, q))(\{A\}, \{A\}) = \bot_L$,

(iii) $(\mathrm{new\_bi\_fun2}(d, q))(\{\{A\}\}, \{\{A\}\}) = \bot_L$,

(iv) $(\mathrm{new\_bi\_fun2}(d, q))(\{A\}, \{\{A\}\}) = (d(q_{\mathbf{1}}, q_{\mathbf{2}}) \sqcup q_{\mathbf{3}}) \sqcap q_{\mathbf{4}}$,

(v) $(\mathrm{new\_bi\_fun2}(d, q))(\{\{A\}\}, \{A\}) = (d(q_{\mathbf{1}}, q_{\mathbf{2}}) \sqcup q_{\mathbf{3}}) \sqcap q_{\mathbf{4}}$, and

(vi) for every element u of A holds $(\mathrm{new\_bi\_fun2}(d, q))(u, \{A\}) = d(u, q_{\mathbf{1}}) \sqcup q_{\mathbf{3}}$ and $(\mathrm{new\_bi\_fun2}(d, q))(\{A\}, u) = d(u, q_{\mathbf{1}}) \sqcup q_{\mathbf{3}}$ and $(\mathrm{new\_bi\_fun2}(d, q))(u, \{\{A\}\}) = d(u, q_{\mathbf{2}}) \sqcup q_{\mathbf{3}}$ and $(\mathrm{new\_bi\_fun2}(d, q))(\{\{A\}\}, u) = d(u, q_{\mathbf{2}}) \sqcup q_{\mathbf{3}}$.

Next we state several propositions:

(10) Let A be a non empty set, L be a lower-bounded lattice, and d be a bifunction from A into L. Suppose d is zeroed. Let q be an element of $[: A, A,$ the carrier of $L,$ the carrier of $L :]$. Then $\mathrm{new\_bi\_fun2}(d, q)$ is zeroed.

(11) Let A be a non empty set, L be a lower-bounded lattice, and d be a bifunction from A into L. Suppose d is symmetric. Let q be an element of $[: A, A,$ the carrier of $L,$ the carrier of $L :]$. Then $\mathrm{new\_bi\_fun2}(d, q)$ is symmetric.

(12) Let A be a non empty set and L be a lower-bounded lattice. Suppose L is modular. Let d be a bifunction from A into L. Suppose d is symmetric and satisfies triangle inequality. Let q be an element of $[: A, A,$ the carrier of $L,$ the carrier of $L :]$. If $d(q_{\mathbf{1}}, q_{\mathbf{2}}) \leqslant q_{\mathbf{3}} \sqcup q_{\mathbf{4}}$, then $\mathrm{new\_bi\_fun2}(d, q)$ satisfies triangle inequality.

(13) For every set A holds $A \subseteq \mathrm{new\_set2}\,A$.

(14) Let A be a non empty set, L be a lower-bounded lattice, d be a bifunction from A into L, and q be an element of $[: A, A,$ the carrier of $L,$ the carrier of $L :]$. Then $d \subseteq \mathrm{new\_bi\_fun2}(d, q)$.

Let A be a non empty set and let O be an ordinal number. The functor $\mathrm{ConsecutiveSet2}(A, O)$ is defined by the condition (Def. 6).

(Def. 6) There exists a transfinite sequence L_0 such that

(i) $\mathrm{ConsecutiveSet2}(A, O) = \mathrm{last}\, L_0$,

(ii) $\mathrm{dom}\, L_0 = \mathrm{succ}\, O$,

(iii) $L_0(\emptyset) = A$,

(iv) for every ordinal number C and for every set z such that $\operatorname{succ} C \in \operatorname{succ} O$ and $z = L_0(C)$ holds $L_0(\operatorname{succ} C) = \text{new\_set2}\, z$, and

(v) for every ordinal number C and for every transfinite sequence L_1 such that $C \in \operatorname{succ} O$ and $C \neq \emptyset$ and C is a limit ordinal number and $L_1 = L_0 \restriction C$ holds $L_0(C) = \bigcup \operatorname{rng} L_1$.

One can prove the following three propositions:

(15) For every non empty set A holds $\text{ConsecutiveSet2}(A, \emptyset) = A$.

(16) For every non empty set A and for every ordinal number O holds $\text{ConsecutiveSet2}(A, \operatorname{succ} O) = \text{new\_set2}\, \text{ConsecutiveSet2}(A, O)$.

(17) Let A be a non empty set, O be an ordinal number, and T be a transfinite sequence. Suppose $O \neq \emptyset$ and O is a limit ordinal number and $\operatorname{dom} T = O$ and for every ordinal number O_1 such that $O_1 \in O$ holds $T(O_1) = \text{ConsecutiveSet2}(A, O_1)$. Then $\text{ConsecutiveSet2}(A, O) = \bigcup \operatorname{rng} T$.

Let A be a non empty set and let O be an ordinal number. Note that $\text{ConsecutiveSet2}(A, O)$ is non empty.

We now state the proposition

(18) For every non empty set A and for every ordinal number O holds $A \subseteq \text{ConsecutiveSet2}(A, O)$.

Let A be a non empty set, let L be a lower-bounded lattice, let d be a bifunction from A into L, let q be a sequence of quadruples of d, and let O be an ordinal number. Let us assume that $O \in \operatorname{dom} q$. The functor $\text{Quadr2}(q, O)$ yielding an element of $[\!:\, \text{ConsecutiveSet2}(A, O), \text{ConsecutiveSet2}(A, O)$, the carrier of L, the carrier of $L\, :\!]$ is defined by:

(Def. 7) $\text{Quadr2}(q, O) = q(O)$.

Let A be a non empty set, let L be a lower-bounded lattice, let d be a bifunction from A into L, let q be a sequence of quadruples of d, and let O be an ordinal number. The functor $\text{ConsecutiveDelta2}(q, O)$ is defined by the condition (Def. 8).

(Def. 8) There exists a transfinite sequence L_0 such that

(i) $\text{ConsecutiveDelta2}(q, O) = \operatorname{last} L_0$,

(ii) $\operatorname{dom} L_0 = \operatorname{succ} O$,

(iii) $L_0(\emptyset) = d$,

(iv) for every ordinal number C and for every set z such that $\operatorname{succ} C \in \operatorname{succ} O$ and $z = L_0(C)$ holds $L_0(\operatorname{succ} C) = \text{new\_bi\_fun2}(\text{BiFun}(z, \text{ConsecutiveSet2}(A, C), L), \text{Quadr2}(q, C))$, and

(v) for every ordinal number C and for every transfinite sequence L_1 such that $C \in \operatorname{succ} O$ and $C \neq \emptyset$ and C is a limit ordinal number and $L_1 = L_0 \restriction C$ holds $L_0(C) = \bigcup \operatorname{rng} L_1$.

Next we state several propositions:

(19) Let A be a non empty set, L be a lower-bounded lattice, d be a bifunction from A into L, and q be a sequence of quadruples of d. Then $\mathrm{ConsecutiveDelta2}(q, \emptyset) = d$.

(20) Let A be a non empty set, L be a lower-bounded lattice, d be a bifunction from A into L, q be a sequence of quadruples of d, and O be an ordinal number. Then $\mathrm{ConsecutiveDelta2}(q, \mathrm{succ}\, O) = \mathrm{new\_bi\_fun2}(\mathrm{BiFun}(\mathrm{ConsecutiveDelta2}(q, O), \mathrm{ConsecutiveSet2}(A, O), L), \mathrm{Quadr2}(q, O))$.

(21) Let A be a non empty set, L be a lower-bounded lattice, d be a bifunction from A into L, q be a sequence of quadruples of d, T be a transfinite sequence, and O be an ordinal number. Suppose $O \neq \emptyset$ and O is a limit ordinal number and $\mathrm{dom}\, T = O$ and for every ordinal number O_1 such that $O_1 \in O$ holds $T(O_1) = \mathrm{ConsecutiveDelta2}(q, O_1)$. Then $\mathrm{ConsecutiveDelta2}(q, O) = \bigcup \mathrm{rng}\, T$.

(22) For every non empty set A and for all ordinal numbers O, O_1, O_2 such that $O_1 \subseteq O_2$ holds $\mathrm{ConsecutiveSet2}(A, O_1) \subseteq \mathrm{ConsecutiveSet2}(A, O_2)$.

(23) Let A be a non empty set, L be a lower-bounded lattice, d be a bifunction from A into L, q be a sequence of quadruples of d, and O be an ordinal number. Then $\mathrm{ConsecutiveDelta2}(q, O)$ is a bifunction from $\mathrm{ConsecutiveSet2}(A, O)$ into L.

Let A be a non empty set, let L be a lower-bounded lattice, let d be a bifunction from A into L, let q be a sequence of quadruples of d, and let O be an ordinal number. Then $\mathrm{ConsecutiveDelta2}(q, O)$ is a bifunction from $\mathrm{ConsecutiveSet2}(A, O)$ into L.

The following propositions are true:

(24) Let A be a non empty set, L be a lower-bounded lattice, d be a bifunction from A into L, q be a sequence of quadruples of d, and O be an ordinal number. Then $d \subseteq \mathrm{ConsecutiveDelta2}(q, O)$.

(25) Let A be a non empty set, L be a lower-bounded lattice, d be a bifunction from A into L, O_1, O_2 be ordinal numbers, and q be a sequence of quadruples of d. If $O_1 \subseteq O_2$, then $\mathrm{ConsecutiveDelta2}(q, O_1) \subseteq \mathrm{ConsecutiveDelta2}(q, O_2)$.

(26) Let A be a non empty set, L be a lower-bounded lattice, and d be a bifunction from A into L. Suppose d is zeroed. Let q be a sequence of quadruples of d and O be an ordinal number. Then $\mathrm{ConsecutiveDelta2}(q, O)$ is zeroed.

(27) Let A be a non empty set, L be a lower-bounded lattice, and d be a bifunction from A into L. Suppose d is symmetric. Let q be a sequence of quadruples of d and O be an ordinal number. Then $\mathrm{ConsecutiveDelta2}(q, O)$ is symmetric.

(28) Let A be a non empty set and L be a lower-bounded lattice. Suppose L is modular. Let d be a bifunction from A into L. Suppose d is symmetric and satisfies triangle inequality. Let O be an ordinal number and q be a sequence of quadruples of d. If $O \subseteq \mathrm{DistEsti}(d)$, then $\mathrm{ConsecutiveDelta2}(q, O)$ satisfies triangle inequality.

(29) Let A be a non empty set, L be a lower-bounded modular lattice, d be a distance function of A, L, O be an ordinal number, and q be a sequence of quadruples of d. If $O \subseteq \mathrm{DistEsti}(d)$, then $\mathrm{ConsecutiveDelta2}(q, O)$ is a distance function of $\mathrm{ConsecutiveSet2}(A, O)$, L.

Let A be a non empty set, let L be a lower-bounded lattice, and let d be a bifunction from A into L. The functor $\mathrm{NextSet2}\, d$ is defined by:

(Def. 9) $\mathrm{NextSet2}\, d = \mathrm{ConsecutiveSet2}(A, \mathrm{DistEsti}(d))$.

Let A be a non empty set, let L be a lower-bounded lattice, and let d be a bifunction from A into L. Note that $\mathrm{NextSet2}\, d$ is non empty.

Let A be a non empty set, let L be a lower-bounded lattice, let d be a bifunction from A into L, and let q be a sequence of quadruples of d. The functor $\mathrm{NextDelta2}\, q$ is defined as follows:

(Def. 10) $\mathrm{NextDelta2}\, q = \mathrm{ConsecutiveDelta2}(q, \mathrm{DistEsti}(d))$.

Let A be a non empty set, let L be a lower-bounded modular lattice, let d be a distance function of A, L, and let q be a sequence of quadruples of d. Then $\mathrm{NextDelta2}\, q$ is a distance function of $\mathrm{NextSet2}\, d$, L.

Let A be a non empty set, let L be a lower-bounded lattice, let d be a distance function of A, L, let A_1 be a non empty set, and let d_1 be a distance function of A_1, L. We say that A_1, d_1 is extension2 of A, d if and only if:

(Def. 11) There exists a sequence q of quadruples of d such that $A_1 = \mathrm{NextSet2}\, d$ and $d_1 = \mathrm{NextDelta2}\, q$.

Next we state the proposition

(30) Let A be a non empty set, L be a lower-bounded lattice, d be a distance function of A, L, A_1 be a non empty set, and d_1 be a distance function of A_1, L. Suppose A_1, d_1 is extension2 of A, d. Let x, y be elements of A and a, b be elements of L. Suppose $d(x, y) \leqslant a \sqcup b$. Then there exist elements z_1, z_2 of A_1 such that $d_1(x, z_1) = a$ and $d_1(z_1, z_2) = (d(x, y) \sqcup a) \sqcap b$ and $d_1(z_2, y) = a$.

Let A be a non empty set, let L be a lower-bounded modular lattice, and let d be a distance function of A, L. A function is called a ExtensionSeq2 of A, d if it satisfies the conditions (Def. 12).

(Def. 12)(i) $\mathrm{dom}\, \mathrm{it} = \mathbb{N}$,

(ii) $\mathrm{it}(0) = \langle A, d \rangle$, and

(iii) for every natural number n there exists a non empty set A' and there exists a distance function d' of A', L and there exists a non empty set

A_1 and there exists a distance function d_1 of A_1, L such that A_1, d_1 is extension2 of A', d' and $\mathrm{it}(n) = \langle A', d' \rangle$ and $\mathrm{it}(n+1) = \langle A_1, d_1 \rangle$.

We now state several propositions:

(31) Let A be a non empty set, L be a lower-bounded modular lattice, d be a distance function of A, L, S be a ExtensionSeq2 of A, d, and k, l be natural numbers. If $k \leqslant l$, then $S(k)_{\mathbf{1}} \subseteq S(l)_{\mathbf{1}}$.

(32) Let A be a non empty set, L be a lower-bounded modular lattice, d be a distance function of A, L, S be a ExtensionSeq2 of A, d, and k, l be natural numbers. If $k \leqslant l$, then $S(k)_{\mathbf{2}} \subseteq S(l)_{\mathbf{2}}$.

(33) Let L be a lower-bounded modular lattice, S be a ExtensionSeq2 of the carrier of L, $\delta_0(L)$, and F_1 be a non empty set. Suppose $F_1 = \bigcup\{S(i)_{\mathbf{1}}$: i ranges over natural numbers$\}$. Then $\bigcup\{S(i)_{\mathbf{2}}$: i ranges over natural numbers$\}$ is a distance function of F_1, L.

(34) Let L be a lower-bounded modular lattice, S be a ExtensionSeq2 of the carrier of L, $\delta_0(L)$, F_1 be a non empty set, F_2 be a distance function of F_1, L, x, y be elements of F_1, and a, b be elements of L. Suppose $F_1 = \bigcup\{S(i)_{\mathbf{1}}$: i ranges over natural numbers$\}$ and $F_2 = \bigcup\{S(i)_{\mathbf{2}}$: i ranges over natural numbers$\}$ and $F_2(x, y) \leqslant a \sqcup b$. Then there exist elements z_1, z_2 of F_1 such that $F_2(x, z_1) = a$ and $F_2(z_1, z_2) = (F_2(x, y) \sqcup a) \sqcap b$ and $F_2(z_2, y) = a$.

(35) Let L be a lower-bounded modular lattice, S be a ExtensionSeq2 of the carrier of L, $\delta_0(L)$, F_1 be a non empty set, F_2 be a distance function of F_1, L, f be a homomorphism from L to $\mathrm{EqRelPoset}(F_1)$, e_1, e_2 be equivalence relations of F_1, and x, y be sets. Suppose that

(i) $f = \alpha(F_2)$,

(ii) $F_1 = \bigcup\{S(i)_{\mathbf{1}} : i$ ranges over natural numbers$\}$,

(iii) $F_2 = \bigcup\{S(i)_{\mathbf{2}} : i$ ranges over natural numbers$\}$,

(iv) $e_1 \in$ the carrier of $\mathrm{Im}\, f$,

(v) $e_2 \in$ the carrier of $\mathrm{Im}\, f$, and

(vi) $\langle x, y \rangle \in e_1 \sqcup e_2$.

Then there exists a non empty finite sequence F of elements of F_1 such that $\mathrm{len}\, F = 2 + 2$ and x and y are joint by F, e_1 and e_2.

(36) For every lower-bounded modular lattice L holds L has a representation of type $\leqslant 2$.

(37) For every lower-bounded lattice L holds L has a representation of type $\leqslant 2$ iff L is modular.

References

[1] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[2] Grzegorz Bancerek. The ordinal numbers. *Formalized Mathematics*, 1(**1**):91–96, 1990.

[3] Grzegorz Bancerek. Complete lattices. *Formalized Mathematics*, 2(**5**):719–725, 1991.

[4] Grzegorz Bancerek. Bounds in posets and relational substructures. *Formalized Mathematics*, 6(**1**):81–91, 1997.

[5] Grzegorz Bancerek. Directed sets, nets, ideals, filters, and maps. *Formalized Mathematics*, 6(**1**):93–107, 1997.

[6] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[7] Józef Białas. Group and field definitions. *Formalized Mathematics*, 1(**3**):433–439, 1990.

[8] Czesław Byliński. Binary operations. *Formalized Mathematics*, 1(**1**):175–180, 1990.

[9] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[10] Czesław Byliński. Functions from a set to a set. *Formalized Mathematics*, 1(**1**):153–164, 1990.

[11] Czesław Byliński. Some basic properties of sets. *Formalized Mathematics*, 1(**1**):47–53, 1990.

[12] Czesław Byliński. Galois connections. *Formalized Mathematics*, 6(**1**):131–143, 1997.

[13] Adam Grabowski. On the category of posets. *Formalized Mathematics*, 5(**4**):501–505, 1996.

[14] George Grätzer. *General Lattice Theory.* Academic Press, New York, 1978.

[15] Jarosław Gryko. The Jónson's theorem. *Formalized Mathematics*, 6(**4**):515–524, 1997.

[16] Adam Naumowicz. On the characterization of modular and distributive lattices. *Formalized Mathematics*, 7(**1**):53–55, 1998.

[17] Beata Padlewska and Agata Darmochwał. Topological spaces and continuous functions. *Formalized Mathematics*, 1(**1**):223–230, 1990.

[18] Konrad Raczkowski and Paweł Sadowski. Equivalence relations and classes of abstraction. *Formalized Mathematics*, 1(**3**):441–444, 1990.

[19] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.

[20] Andrzej Trybulec. Tuples, projections and Cartesian products. *Formalized Mathematics*, 1(**1**):97–105, 1990.

[21] Wojciech A. Trybulec. Partially ordered sets. *Formalized Mathematics*, 1(**2**):313–319, 1990.

[22] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.

[23] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

[24] Edmund Woronowicz. Relations defined on sets. *Formalized Mathematics*, 1(**1**):181–186, 1990.

[25] Mariusz Żynel and Czesław Byliński. Properties of relational structures, posets, lattices and maps. *Formalized Mathematics*, 6(**1**):123–130, 1997.

Received June 29, 2000

alalalalalal alalalalalalal

Index of MML Identifiers

Contents

Formaliz. Math. 9 (**2**)

Continued on inside back cover

Continued on inside back cover