Static Analysis By Abstract Interpretation

Jan Peleska[1]

Centre of Information Technology
University of Bremen
Germany

Helge Löding[2]

GESy Graduate School of Embedded Systems
University of Bremen
Germany

1 Theoretical Foundations

1.1 Lattices and Galois Connections

Recall that a binary relation $\sqsubseteq$ on a set L is called a (partial) order if $\sqsubseteq$ is reflexive, transitive and anti-symmetric. An element $y \in L$ is called an *upper bound of* $X \subseteq L$ if $x \sqsubseteq y$ holds for all $x \in X$. The lower bound of a set is defined dually. An upper bound y' of X is called a *least upper bound of* X and denoted by $\sqcup X$ if $y' \sqsubseteq y$ holds for all upper bounds y of X. Dually, the *greatest lower bound* $\sqcap X$ of a set X is defined.

An ordered set $(L, \sqsubseteq)$ is called a *complete lattice*, if $\sqcap X$ and $\sqcup X$ exist for all subsets $X \subseteq L$. Lattice L has a *largest element* (or *top*) denoted by $\top = \sqcup L$ and a *smallest element* (or *bottom*) denoted by $\bot = \sqcap L$. Least upper bounds and greatest lower bounds induce binary operations $\sqcup, \sqcap : L \times L \to L$ by defining $x \sqcup y =_{\text{def}} \sqcup\{x, y\}$ (the *join* of x and y) and $x \sqcap y =_{\text{def}} \sqcap\{x, y\}$ (the *meet* of x and y), respectively. If the join and meet are well-defined for an ordered set $(L, \sqsubseteq)$ but $\sqcup X, \sqcap X$ do not exist for all $X \subseteq L$ then $(L, \sqsubseteq)$ is called an *(incomplete) lattice*.

Example 1.1 (i) For every set M the *power set lattice* is defined by $(\mathbf{P}(M), \subseteq)$. The join is defined by $m \sqcup m' =_{\text{def}} m \cup m'$, the meet by $m \sqcap m' =_{\text{def}} m \cap m'$. Top and bottom elements are $\top = M$, $\bot = \varnothing$, respectively.

[1] Email:jp@tzi.de
[2] Email:hloeding@tzi.de

(ii) For every set M wie can introduce a nearly trivial ordering $\sqsubseteq$ by adding two new elements $\top, \bot \notin M$ and defining a lattice $(M \cup \{\top, \bot\}, \sqsubseteq)$ such that all $m \neq m' \in M$ are incomparable and $\forall m \in M : \bot \sqsubseteq m \sqsubseteq \top$.

(iii) Applying the construction (ii) to Booleans $\mathbb{B} = \{\texttt{false}, \texttt{true}\}$ results in the lattice $(L(\mathbb{B}), \sqsubseteq)$ with $L(\mathbb{B}) =_{\text{def}} \{\bot, \texttt{false}, \texttt{true}, \top\}$, $\bot \sqsubseteq \texttt{false} \sqsubseteq \top, \bot \sqsubseteq \texttt{true} \sqsubseteq \top$ and $\texttt{true}, \texttt{false}$ incomparable.

(iv) $(\mathbb{Q}, \leq)$ is an *incomplete* lattice: Take any infinite set $S \subseteq \mathbb{Q}$ whose elements are converging towards a transcendent number, say $\sqrt{2}$, from below. Then $\sqcup S \notin \mathbb{Q}$.

(v) The lattice of *intervals* over reals including $\pm\infty$ is defined as $(\mathbb{R}, \subseteq)$ with $[\underline{a}, \overline{a}] \sqcap [\underline{b}, \overline{b}] =_{\text{def}} [\underline{a}, \overline{a}] \cap [\underline{b}, \overline{b}]$ and $[\underline{a}, \overline{a}] \sqcup [\underline{b}, \overline{b}] =_{\text{def}} [min\{\underline{a}, \underline{b}\}, max\{\overline{a}, \overline{b}\}]$. The join of $[\underline{a}, \overline{a}]$ and $[\underline{b}, \overline{b}]$ is also called the *interval hull* of $[\underline{a}, \overline{a}]$ and $[\underline{b}, \overline{b}]$. The maximal element is $\top = [-\infty, +\infty]$, $\bot = [] = \varnothing$.

From the collection of canonic ways to construct new lattices from existing ones $(L, \sqsubseteq), (L_1, \sqsubseteq_1), (L_2, \sqsubseteq_2)$, we need (1) cross products $(L_1 \times L_2, \sqsubseteq')$ where the partial order is defined by $(x_1, x_2) \sqsubseteq' (y_1, y_2)$ if and only if $x_1 \sqsubseteq_1 y_1 \wedge x_2 \sqsubseteq_2 y_2$ and (2) partial function spaces $(V \nrightarrow L, \sqsubseteq')$ where $f \sqsubseteq' g$ for $f, g \in V \nrightarrow L$ if and only if $\text{dom } f \subseteq \text{dom } g \wedge (\forall x \in \text{dom } f : f(x) \sqsubseteq g(x))$.

Mappings $\phi : (L_1, \sqsubseteq_1) \to (L_2, \sqsubseteq_2)$ between ordered sets are called *monotone* if $x \sqsubseteq_1 y$ implies $\phi(x) \sqsubseteq_2 \phi(y)$ for all $x, y \in L$. Mappings $\phi : (L_1, \sqsubseteq_1) \to (L_2, \sqsubseteq_2)$ between lattices are called *homomorphisms* if they respect meets and joins, that is, $\phi(x \sqcup_1 y) = \phi(x) \sqcup_2 \phi(y)$ and $\phi(x \sqcap_1 y) = \phi(x) \sqcap_2 \phi(y)$ for all $x, y \in (L_1, \sqsubseteq_1)$. Since $x \sqsubseteq_1 y$ implies $x \sqcup_1 y = y$ and $x \sqcap_1 y = x$, homomorphisms are monotone.

A *Galois connection (GC)* between lattices $(L_1, \sqsubseteq_1), (L_2, \sqsubseteq_2)$ is a tuple of mappings $\_^{\triangleright} : (L_1, \sqsubseteq_1) \to (L_2, \sqsubseteq_2)$ (called *right*) and $\_^{\triangleleft} : (L_2, \sqsubseteq_2) \to (L_1, \sqsubseteq_1)$ (called *left*) such that

$$a^{\triangleright} \sqsubseteq_2 b \Leftrightarrow a \sqsubseteq_1 b^{\triangleleft} \qquad (Gal)$$

for all $a \in L_1, b \in L_2$. This defining property implies that Galois connections satisfy for all $p, p_1, p_2 \in L_1$, $q, q_1, q_2 \in L_2$:

(Gal1) $p \sqsubseteq_1 p^{\triangleright\triangleleft}$ and $q^{\triangleleft\triangleright} \sqsubseteq_2 q$

(Gal2) $\_^{\triangleright}$ and $\_^{\triangleleft}$ are monotone: $p_1 \sqsubseteq_1 p_2 \Rightarrow {p_1}^{\triangleright} \sqsubseteq_2 {p_2}^{\triangleright}$ and $q_1 \sqsubseteq_2 q_2 \Rightarrow {q_1}^{\triangleleft} \sqsubseteq_1 {q_2}^{\triangleleft}$.

(Gal3) $p^{\triangleright} = p^{\triangleright\triangleleft\triangleright}$ and $q^{\triangleleft} = q^{\triangleleft\triangleright\triangleleft}$

In [1, 7.26] it is shown that (Gal1), (Gal2), (Gal3) are indeed equivalent to the defining property (Gal). A GC is called *exact* if (Gal1) holds with $q^{\triangleleft\triangleright} = q$. Intuitively speaking, exact GCs are of particular interest for abstract interpretation, because an abstract state element q is always mapped to the "most general" concrete state element $p = q^{\triangleleft}$ which is still abstracted back to q. For GCs between complete lattices, the right mapping of the GC preserves suprema [1, 7.31]:

$$\forall Z \subseteq L_1 : \left(\bigsqcup Z\right)^{\triangleright} = \bigsqcup (Z^{\triangleright})$$

where $Z^{\triangleright}$ is short for $\{z^{\triangleright} \mid z \in Z\}$.

Given a GC as above, we can *lift* operations, e.g., $\diamond : L_1 \times L_1 \to L_1$ to abstracted

operations $[\diamond] : L_2 \times L_2 \to L_2$ according to the rule

$$x[\diamond]y =_{\text{def}} (x^{\triangleleft} \diamond y^{\triangleleft})^{\triangleright}$$

For $n$ary operations and operations with other ranges (e.g. the lattice over Booleans), analogous rules exist.

For any concrete datatype t used in a programming language and operation $\diamond : t \times t \to t$, the operation is defined on the associated power set lattice by

$$\diamond_{\mathbf{P}} : \mathbf{P}(t) \times \mathbf{P}(t) \to \mathbf{P}(t)$$
$$m \diamond_{\mathbf{P}} m' =_{\text{def}} \{x \diamond x' \mid x \in m \wedge x' \in m'\}$$

Lemma 1.2 *Given a Galois connection $\mathbf{P}(t) \underset{\triangleright}{\overset{\triangleleft}{\leftrightarrows}} L$ between the power set lattice of a datattype t and a complete lattice L and an operation $\diamond : t \times t \to t$. Then $\diamond$ can be lifted to $\diamond_L : L \times L \to L$ by*

$$a \diamond_L b = \bigsqcup \{\{x \diamond y\}^{\triangleright} \mid x \in a^{\triangleleft} \wedge y \in b^{\triangleleft}\}$$

Proof. According to the lifting of operations between lattices introduced above we calculate

$$\begin{aligned} a \diamond_L b =_{\text{def}} & (a^{\triangleleft} \diamond_{\mathbf{P}} b^{\triangleleft})^{\triangleright} \\ = & \{x \diamond y \mid x \in a^{\triangleleft} \wedge y \in b^{\triangleleft}\}^{\triangleright} \\ = & \left(\bigsqcup \{\{x \diamond y\} \mid x \in a^{\triangleleft} \wedge y \in b^{\triangleleft}\}\right)^{\triangleright} \\ = & \bigsqcup \{\{x \diamond y\}^{\triangleright} \mid x \in a^{\triangleleft} \wedge y \in b^{\triangleleft}\} \end{aligned}$$

□

Lemma 1.2 can be generalised in the obvious way to n-ary operations:

Corollary 1.3 *Given GCs $\mathbf{P}(t_i) \underset{\triangleright}{\overset{\triangleleft}{\leftrightarrows}} L_i, i = 0, \ldots, n$ between the power set lattices $\mathbf{P}(t_i)$ of datatypes t_i and complete lattices L_i and an n-ary operation $\omega : t_1 \times \ldots \times t_n \to t_0$. Then ω can be lifted to $\omega_L : L_1 \times \ldots \times L_n \to L_0$ by*

$$\omega_L(a_1, \ldots, a_n) = \bigsqcup \{\{\omega(x_1, \ldots, x_n)\}^{\triangleright} \mid \forall i \in \{1, \ldots, n\} : x_i \in {a_i}^{\triangleleft}\}$$

Example 1.4 Let us lift the +-operation to the interval lattice. First we define a GC between $(\mathbf{P}(float), \subseteq)$ and $(\mathbb{IR}, \subseteq)$ via

$$m^{\triangleright} =_{\text{def}} [inf(m), sup(m)]$$
$$[\underline{a}, \overline{a}]^{\triangleleft} =_{\text{def}} [\underline{a}, \overline{a}]$$

Now we lift the +-operation according to the recipe above and define

$$\begin{aligned} [\underline{a}, \overline{a}][+][\underline{b}, \overline{b}] =_{\text{def}} & \\ & \{x + y \mid x \in [\underline{a}, \overline{a}] \wedge y \in [\underline{b}, \overline{b}]\}^{\triangleright} = \\ & \bigsqcup \{[x + y, x + y] \mid x \in [\underline{a}, \overline{a}] \wedge y \in [\underline{b}, \overline{b}]\} = \\ & [\underline{a} + \underline{b}, \overline{a} + \overline{b}] \end{aligned}$$

1.2 Construction of Abstract Interpretation Semantics

Given any transition system $TS = (S, S_0, \longrightarrow)$ the most fine-grained state space abstraction possible is represented by the power set lattice $L_{\mathbf{P}}(S) = (\mathbf{P}(S), \subseteq)$ with join operation $\cup$ and meet $\cap$. We introduce an abstract interpretation semantics on $L_{\mathbf{P}}(S)$ by turning it into a state transition system $TS_{\mathbf{P}} = (L_{\mathbf{P}}(S), \{S_0\}, \longrightarrow_{\mathbf{P}})$ by lifting the original transition relation to sets: Using Plotkin-style notation, this can be specified as

$$\frac{\forall i \in I, s_i, s_i' \in S : s_i \longrightarrow s_i'}{\{s_i \mid i \in I\} \longrightarrow_{\mathbf{P}} \{s_i' \mid i \in I\}} \qquad (P1)$$

We extend this definition by two additional rules allowing terminal states of S to transit to the empty set in $\mathbf{P}(S)$:

$$\frac{A \subseteq S,\ \forall s \in A, s' \in S : s \not\longrightarrow s'}{A \longrightarrow_{\mathbf{P}} \varnothing} \qquad (P2)$$

$$\frac{}{\varnothing \longrightarrow_{\mathbf{P}} \varnothing} \qquad (P3)$$

Compared to the original transition system TS, this abstract interpretation $\longrightarrow_{\mathbf{P}}$ introduces no loss of information, since its restriction to pairs of singleton sets is equivalent to the original transition relation:

$$\forall s_1, s_2 \in S : s_1 \longrightarrow s_2 \Leftrightarrow \{s_1\} \longrightarrow_{\mathbf{P}} \{s_2\}$$

It is, however, an abstraction, since for transitions between states with cardinality higher than one, say $\{s_1, s_2, \ldots\} \longrightarrow_{\mathbf{P}} \{s_1', s_2', \ldots\}$, only the possible resulting states are listed $(s_1', s_2', \ldots)$ but the information whether, for example, $s_1 \longrightarrow s_1'$ or $s_1 \longrightarrow s_2'$ is no longer available.

Observe that $\longrightarrow_{\mathbf{P}}$ obviously satisfies

Lemma 1.5 $\forall q, q', r \subseteq S : q \longrightarrow_{\mathbf{P}} q' \wedge r \subseteq q \Rightarrow (\exists r' \subseteq q' : r \longrightarrow_{\mathbf{P}} r')$

Now, given any other transition system $TS_L = (L, L_0, \longrightarrow_L)$ based on a lattice $(L, \sqsubseteq)$ we can check whether TS_L is a valid abstract interpretation of TS by the aid of $TS_{\mathbf{P}}$ and Galois connections:

Definition 1.6 Transition system $TS_L = (L, L_0, \longrightarrow_L)$, based on a lattice $(L, \sqsubseteq)$, is a *valid abstract interpretation of* $TS = (S, S_0, \longrightarrow)$ if

(i) There exists a Galois connection
$^{\triangleright} : (\mathbf{P}(S), \subseteq) \to (L, \sqsubseteq)$, $^{\triangleleft} : (L, \sqsubseteq) \to (\mathbf{P}(S), \subseteq)$

(ii) The transition relation $\longrightarrow_L$ is a *valid abstract relation* in the following sense:
$$\forall a, a', b \in L : (a \longrightarrow_L a' \wedge b \sqsubseteq a \Rightarrow \exists b' \in L : b \longrightarrow_L b' \wedge b' \sqsubseteq a')$$

(iii) The transition relation $\longrightarrow_L$ satisfies
$$\forall (p, p') \in \longrightarrow_{\mathbf{P}} : \exists a' \in L : p^{\triangleright} \longrightarrow_L a' \wedge p'^{\triangleright} \sqsubseteq a'$$

(iv) The transition relation $\longrightarrow_L$ satisfies
$$\forall (a, a') \in \longrightarrow_L : \exists p' \in \mathbf{P}S : a^{\triangleleft} \longrightarrow_{\mathbf{P}} p' \wedge p' \subseteq a'^{\triangleleft}$$

The following theorem provides a "recipe" for constructing valid abstract interpretations, as soon as a GC according to Definition 1.6, (i) has been established:

Theorem 1.7 *Given lattice* $(L, \sqsubseteq)$ *and Galois connection* $^{\triangleright} : (\mathbf{P}(S), \subseteq) \to (L, \sqsubseteq)$, $^{\triangleleft} : (L, \sqsubseteq) \to (\mathbf{P}(S), \subseteq)$, *define transition system* $TS_L = (L, L_0, \longrightarrow_L)$ *by*

(i) $$L_0 = \{S_0\}^{\triangleright}$$

(ii) $$\frac{p^{\triangleright\triangleleft} \longrightarrow_{\mathbf{P}} p'}{p^{\triangleright} \longrightarrow_L p'^{\triangleright}}$$

(iii) $$\frac{a^{\triangleleft} \longrightarrow_{\mathbf{P}} p'}{a \longrightarrow_L p'^{\triangleright}}$$

Then TS_L *is a valid abstract interpretation of* TS *in the sense of Definition 1.6.*

Proof. We first show that $\longrightarrow_L$ satisfies condition (ii) of Definition 1.6: Suppose $a \longrightarrow_L a'$ and $b \sqsubseteq a$. Since $^{\triangleleft}$ is monotone, we conclude $b^{\triangleleft} \subseteq a^{\triangleleft}$. Since transition $a \longrightarrow_L a'$ exists either due to rule (ii) or due to rule (iii) of the theorem, the existence of $p' \in \mathbf{P}(S)$ such that $a^{\triangleleft} \longrightarrow_{\mathbf{P}} p'$, $a \longrightarrow_L p'^{\triangleright}$ and $a' = p'^{\triangleright}$ follows for both cases. Then $b^{\triangleleft} \subseteq a^{\triangleleft}$ and Lemma 1.5 imply existence of p'' such that $b^{\triangleleft} \longrightarrow_{\mathbf{P}} p''$ and $p'' \subseteq p'$. Applying rule (iii) to b, p'' we conclude that $b \longrightarrow_L b'$. Moreover, monotonicity of $^{\triangleright}$ and $p'' \subseteq p'$ imply $b' \sqsubseteq p'^{\triangleright} = a'$.

Next, condition (iii) of Definition 1.6 is established: Suppose $p \longrightarrow_{\mathbf{P}} p'$. Set $a =_{\text{def}} p^{\triangleright}$. Then $p \subseteq a^{\triangleleft} = p^{\triangleright\triangleleft}$ because of (Gal1). As a consequence, there exists a superset $\overline{p}$ of p' such that $a^{\triangleleft} \longrightarrow_{\mathbf{P}} \overline{p}$. Condition (iii) of the theorem now implies $a \longrightarrow_L \overline{p}^{\triangleright}$, and monotonicity of $^{\triangleright}$ implies $p'^{\triangleright} \sqsubseteq \overline{p}^{\triangleright}$. Setting $a' =_{\text{def}} \overline{p}^{\triangleright}$ now shows validity of condition (iii) of Definition 1.6.

Finally, the validity of condition (iv) of Definition 1.6 is shown: Suppose $a \longrightarrow_L a'$. Case 1: $\exists p, p' \in \mathbf{P}(S) : p^{\triangleright} = a \wedge p'^{\triangleright} = a' \wedge p^{\triangleright\triangleleft} \longrightarrow_{\mathbf{P}} p'$. Then $a'^{\triangleleft} = p'^{\triangleright\triangleleft}$, so $p' \subseteq a'^{\triangleleft}$ due to (Gal2). Case 2: $\exists p' \in \mathbf{P}(p) : a^{\triangleleft} \longrightarrow_{\mathbf{P}} p' \wedge p'^{\triangleright} = a'$. Then (Gal) implies $p' \subseteq a'^{\triangleleft}$, so the validity of condition (iv) of Definition 1.6 is established once more.

This completes the proof of the theorem. □

Observations.

The conditions of Definition 1.6 and Theorem 1.7 are *not* equivalent. Indeed, Theorem 1.7 specifies the *most exact* transition relation of all relations which are valid in the sense of Definition 1.6. To see this, suppose that the requirements of Definition 1.6 hold and that $p^{\triangleright\triangleleft} \longrightarrow_{\mathbf{P}} p'$ for some $p, p' \in \mathbf{P}(S)$. Applying condition (iii) of the definition yields existence of $a' \in L$ such that $p^{\triangleright\triangleleft\triangleright} \longrightarrow_L a' \wedge p'^{\triangleright} \sqsubseteq a'$. Then application of (Gal3) implies $p^{\triangleright} \longrightarrow_L a' \wedge p'^{\triangleright} \sqsubseteq a'$. This means that Definition 1.6 only guarantees existence of a transition to a target state a' with $p'^{\triangleright} \sqsubseteq a'$. In other words, the condition (ii) of Theorem 1.7 is the most exact transition rule admissible according to Definition 1.6, where a' always *coincides* with $p'^{\triangleright}$, instead of just being a lattice member "somewhere above" $p'^{\triangleright}$.

Now suppose that the requirements of Definition 1.6 hold and that $a^{\triangleleft} \longrightarrow_{\mathbf{P}} p'$ for some $a \in L, p' \in \mathbf{P}(S)$. Additionally assume that the GC is exact, that is, $a^{\triangleleft\triangleright} = a$ holds for all $a \in L$. Condition (iii) of Definition 1.6 implies existence of $a' \in L$ such that $a^{\triangleleft\triangleright} \longrightarrow_L a' \wedge p'^{\triangleright} \sqsubseteq a'$. Due to exactness, this just means $a \longrightarrow_L a' \wedge p'^{\triangleright} \sqsubseteq a'$. As a consequence, Definition 1.6 yields existence of a transition to a target state a' with $p'^{\triangleright} \sqsubseteq a'$ in presence of an exact GC. Again, Theorem 1.7 is more precise than

the definition because rule (iii) of the theorem always defines the transition to the target state $a' = p'^{\triangleright}$.

1.3 Construction Principle for Abstract Program Interpretations – Fine-Grained Data Abstraction

For programming languages like C, C++ or Java, the state space S is typically structured into *control locations* $c \in Loc$ and *variable valuations*

$$\sigma \in \Sigma,\ \Sigma =_{\text{def}} X \not\to D,$$
$$D =_{\text{def}} \bigcup_{x \in X} D_x$$
$$\forall \sigma \in \Sigma, x \in \text{dom } \sigma : \sigma(x) \in D_x$$

State transitions are defined explicitly on pairs of control locations and valuations, such as $(c_1, \sigma_1) \longrightarrow (c_2, \sigma_2)$. As a consequence, the associated power set lattice is structured as

$$L_{\mathbf{P}} =_{\text{def}} (\mathbf{P}(Loc \times \Sigma), \{S_0\}, \longrightarrow_{\mathbf{P}}, \subseteq)$$
$$S_0 \subseteq Loc \times \Sigma$$

with operational rule

$$\frac{\forall i \in I, (c_i, \sigma_i), (c_i', \sigma_i') \in Loc \times \Sigma : (c_i, \sigma_i) \longrightarrow (c_i', \sigma_i')}{\{(c_i, \sigma_i) \mid i \in I\} \longrightarrow_{\mathbf{P}} \{(c_i', \sigma_i') \mid i \in I\}}$$

replacing (P1) above.

We will now present a "recipe" for constructing valid abstract interpretations based on the abstraction of datatypes D_x used in the program:

(i) For each D_x used in the program, associate the desired *data type abstraction lattice* $L(D_x)$, so that $L(D_x)$ is *admissible* in the sense that it can be related to the power set lattice via Galois connection

$$^{\triangleright} : (\mathbf{P}(D_x), \subseteq) \to (L(D_x), \sqsubseteq),\ ^{\triangleleft} : (L(D_x), \sqsubseteq) \to (\mathbf{P}(D_x), \subseteq)$$

(ii) Define *abstract valuations*

$$\lambda \in \Sigma_L,\ \Sigma_L =_{\text{def}} X \not\to L(D)$$
$$L(D) = \bigcup_{x \in X} L(D_x)$$
$$\forall \lambda \in \Sigma_L, x \in \text{dom } \lambda : \lambda(x) \in L(D_x)$$

(iii) Define *power set abstraction lattice over abstracted datatypes* $L(D)$:

$$L = (\mathbf{P}(Loc \times \Sigma_L), \subseteq)$$

(iv) Define a GC between $L_{\mathbf{P}}$ and L via

$$\begin{aligned}
&{}^{\triangleright} : (\mathbf{P}(Loc \times \Sigma), \subseteq) \to (\mathbf{P}(Loc \times \Sigma_L), \subseteq)\\
&{}^{\triangleleft} : (\mathbf{P}(Loc \times \Sigma_L), \subseteq) \to (\mathbf{P}(Loc \times \Sigma), \subseteq)\\
&S^{\triangleright} =_{\text{def}} \{(c, \lambda) \mid \exists \sigma \in \Sigma : (c, \sigma) \in S \wedge\\
&\qquad \text{dom } \sigma = \text{dom } \lambda \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\}^{\triangleright} = \lambda(x))\}\\
&U^{\triangleleft} =_{\text{def}} \{(c, \sigma) \mid \exists \lambda \in \Sigma_L : (c, \lambda) \in U \wedge\\
&\qquad \text{dom } \sigma = \text{dom } \lambda \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \lambda(x)^{\triangleleft})\}
\end{aligned}$$

Lemma 1.9 below proves the GC properties for these mappings.

(v) Define transition relation $\longrightarrow_L \subseteq (\mathbf{P}(Loc \times \Sigma_L) \times \mathbf{P}(Loc \times \Sigma_L))$ according to the conditions of Theorem 1.7.

(vi) Together, Lemma 1.9 and Theorem 1.7 imply that we have constructed a valid abstract interpretation in the sense of Definition 1.6.

Lemma 1.8 *The Galois connection defined in (iv) above satisfies for all* $S \in \mathbf{P}(Loc \times \Sigma)$

(i) $S^{\triangleright} = \{(c, \{x \mapsto \{\sigma(x)\}^{\triangleright} \mid x \in dom\ \sigma\}) \mid (c, \sigma) \in S\}$

(ii) $S^{\triangleright\,\triangleleft} = \{(c, \sigma) \mid \exists (c, \sigma') \in S : dom\ \sigma = dom\ \sigma' \wedge (\forall x \in dom\ \sigma : \sigma(x) \in \{\sigma'(x)\}^{\triangleright\,\triangleleft})\}$

Proof. (i) is just an alternative representation of the definition of $S^{\triangleright}$ given in (iv). In order to prove (ii), we calculate

$$\begin{aligned}
S^{\triangleright\,\triangleleft} &= \{(c, \{x \mapsto \{\sigma(x)\}^{\triangleright} \mid x \in \text{dom } \sigma\}) \mid (c, \sigma) \in S\}^{\triangleleft}\\
&= \{(c, \sigma) \mid \exists (c, \sigma') \in S : \text{dom } \sigma = \text{dom } \sigma' \wedge (\forall x \in \text{dom } \sigma : \sigma(x) \in \{\sigma'(x)\}^{\triangleright\,\triangleleft})\}
\end{aligned}$$

□

Lemma 1.9 *Mappings* ${}^{\triangleright}, {}^{\triangleleft}$*, as defined in (iv) above, represent a Galois connection.*

Proof. 1. The definition of ${}^{\triangleright}, {}^{\triangleleft}$ in (iv) immediately implies that both mappings are monotone with respect to partial order $\subseteq$.

2. Suppose $S \in \mathbf{P}(Loc \times \Sigma)$ and $S^{\triangleright} \subseteq U \in \mathbf{P}(Loc \times \Sigma_L)$. We prove that $S \subseteq U^{\triangleleft}$. The definition of ${}^{\triangleright}$, Lemma 1.8 and the fact that $S^{\triangleright} \subseteq U$ imply

$$\begin{aligned}
&U' =_{\text{def}} S^{\triangleright} \subseteq U\\
&U' = \{(c, \{x \mapsto \{\sigma(x)\}^{\triangleright} \mid x \in \text{dom } \sigma\}) \mid (c, \sigma) \in S\}
\end{aligned}$$

Therefore monotonicity of ${}^{\triangleleft}$ implies $U'^{\triangleleft} \subseteq U^{\triangleleft}$ and it is sufficient to prove $S \subseteq U'^{\triangleleft}$. To this end, Lemma 1.8 (ii) implies that

$$\begin{aligned}
U'^{\triangleleft} &= \{(c, \sigma) \mid \exists (c, \sigma') \in S : \text{dom } \sigma = \text{dom } \sigma' \wedge (\forall x \in \text{dom } \sigma : \sigma(x) \in \{\sigma'(x)\}^{\triangleright\,\triangleleft})\}\\
&= \{(c, \sigma) \mid \exists (c, \sigma') \in S : \text{dom } \sigma = \text{dom } \sigma' \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \{\sigma'(x)\}^{\triangleright\,\triangleleft})\}\\
&= \{(c, \sigma) \mid \exists (c, \sigma') \in S : \text{dom } \sigma = \text{dom } \sigma' \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\}^{\triangleright} \subseteq \{\sigma'(x)\}^{\triangleright})\}\\
&\supseteq \{(c, \sigma) \mid \exists (c, \sigma') \in S : \text{dom } \sigma = \text{dom } \sigma' \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\}^{\triangleright} = \{\sigma'(x)\}^{\triangleright})\}\\
&\supseteq S
\end{aligned}$$

so this shows 2.

3. Suppose $S \in \mathbf{P}(Loc \times \Sigma)$ and $U \in \mathbf{P}(Loc \times \Sigma_L)$ and $S \subseteq U^{\triangleleft}$. We prove that $S^{\triangleright} \subseteq U$.

$$
\begin{array}{l}
S \subseteq U^{\triangleleft} \Rightarrow \\
\quad \forall (c,\sigma) \in S : \\
\quad\quad (c,\sigma) \in \{(c',\sigma') \mid \exists \lambda' \in \Sigma_L : (c,\lambda') \in U \wedge \\
\quad\quad \mathrm{dom}\ \sigma' = \mathrm{dom}\ \lambda' \wedge (\forall x \in \mathrm{dom}\ \sigma' : \{\sigma'(x)\} \subseteq \lambda'(x)^{\triangleleft})\} \Rightarrow \\
\quad \forall (c,\sigma) \in S : \exists \lambda \in \Sigma_L : (c,\lambda) \in U \wedge \mathrm{dom}\ \sigma = \mathrm{dom}\ \lambda \wedge \\
\quad\quad (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\} \subseteq \lambda(x)^{\triangleleft}) \Rightarrow \\
\quad \forall (c,\sigma) \in S : \exists \lambda \in \Sigma_L : (c,\lambda) \in U \wedge \mathrm{dom}\ \sigma = \mathrm{dom}\ \lambda \wedge \\
\quad\quad (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\}^{\triangleright} \sqsubseteq \lambda(x)) \Rightarrow \\
\quad \{(c,\lambda) \mid \exists \sigma \in \Sigma : (c,\sigma) \in S \wedge \mathrm{dom}\ \sigma = \mathrm{dom}\ \lambda \wedge \\
\quad\quad (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\}^{\triangleright} \sqsubseteq \lambda(x))\} \subseteq U \Rightarrow \\
\quad \{(c,\lambda) \mid \exists \sigma \in \Sigma : (c,\sigma) \in S \wedge \mathrm{dom}\ \sigma = \mathrm{dom}\ \lambda \wedge \\
\quad\quad (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\}^{\triangleright} = \lambda(x))\} \subseteq U \Rightarrow \\
\quad S^{\triangleright} \subseteq U
\end{array}
$$

□

1.4 *Construction Principle for Abstract Program Interpretations – Coarse-Grained Data Abstraction*

In this section we define another abstraction which is coarser than, but closely related to the the lattice $(\mathbf{P}(Loc \times \Sigma_L), \subseteq)$ defined in the previous section for arbitrary admissible data type abstractions $L(D_x)$. We wish to relate this new abstraction directly to $(\mathbf{P}(Loc \times \Sigma_L), \subseteq)$, without having to go back to the powerset lattice $(\mathbf{P}(Loc \times \Sigma), \subseteq)$. The following theorem shows that this is possible.

Theorem 1.10 *Let $(L_i, \sqsubseteq_i), i = 1, 2$ lattices with Galois connections $^{\triangleright} : (\mathbf{P}(S), \subseteq) \to (L_1, \sqsubseteq_1)$, $^{\triangleleft} : (L_1, \sqsubseteq_1) \to (\mathbf{P}(S), \subseteq)$ and $^{\triangleright'} : (L_1, \sqsubseteq_1) \to (L_2, \sqsubseteq_2)$, $^{\triangleleft'} : (L_2, \sqsubseteq_2) \to (L_1, \sqsubseteq_1)$. Suppose that $\longrightarrow_{L_1}$ has been defined according to Theorem 1.7 by*

$$\frac{p^{\triangleright\triangleleft} \longrightarrow_{\mathbf{P}} p'}{p^{\triangleright} \longrightarrow_{L} p'^{\triangleright}}, \qquad \frac{a_1{}^{\triangleleft} \longrightarrow_{\mathbf{P}} p'}{a_1 \longrightarrow_{L} p'^{\triangleright}}$$

Define $\longrightarrow_{L_2} \subseteq L_2 \times L_2$ by

$$\frac{a_1{}^{\triangleright'\triangleleft'} \longrightarrow_{L_1} a_1'}{a_1{}^{\triangleright'} \longrightarrow_{L_2} a_1'^{\triangleright'}}, \qquad \frac{a_2{}^{\triangleleft'} \longrightarrow_{L_1} a_1'}{a_2 \longrightarrow_{L_2} a_1'^{\triangleright'}}$$

Then $(L_2, \sqsubseteq_2, \longrightarrow_{L_2})$ is a valid abstract interpretation in in the sense of Definition 1.6.

Proof. 1. The composition of GCs is again a GC. This can be checked by proving the validity of (Gal):

$$p \sqsubseteq (b_2{}^{\triangleleft'})^{\triangleleft} \Leftrightarrow p^{\triangleright} \sqsubseteq b_2{}^{\triangleleft'} \Leftrightarrow p^{\triangleright\triangleright'} \sqsubseteq b_2$$

2. We show that the preconditions of the theorem imply that

$$\frac{p_1{}^{\triangleright\triangleright'\triangleleft'\triangleleft} \longrightarrow_{\mathbf{P}} p_1'}{p_1{}^{\triangleright\triangleright'} \longrightarrow_{L_2} p_1'^{\triangleright\triangleright'}}, \qquad \frac{a_2{}^{\triangleleft'\triangleleft} \longrightarrow_{\mathbf{P}} p_1'}{a_2 \longrightarrow_{L_2} p_1'^{\triangleright\triangleright'}} \qquad (*)$$

Then we can apply Theorem 1.7 to the GC $(^{\triangleright\triangleright'}, ^{\triangleleft'\triangleleft})$ and this yields validity of the abstract interpretation $(L_2, \sqsubseteq_2, \longrightarrow_{L_2})$ in in the sense of Definition 1.6. To establish (*) we calculate

$$p_1{}^{\triangleright\triangleright'\triangleleft'\triangleleft} \longrightarrow_{\mathbf{P}} p_1' \Rightarrow (p_1{}^{\triangleright\triangleright'\triangleleft'})^{\triangleleft} \longrightarrow_{\mathbf{P}} p_1'$$
$$\Rightarrow p_1{}^{\triangleright\triangleright'\triangleleft'} \longrightarrow_{L_1} p_1'{}^{\triangleright}$$
$$\Rightarrow p_1{}^{\triangleright\triangleright'} \longrightarrow_{L_2} p_1'{}^{\triangleright\triangleright'}$$

and

$$a_2{}^{\triangleleft'\triangleleft} \longrightarrow_{\mathbf{P}} p_1' \Rightarrow (a_2{}^{\triangleleft'})^{\triangleleft} \longrightarrow_{\mathbf{P}} p_1'$$
$$\Rightarrow a_2{}^{\triangleleft'} \longrightarrow_{L_1} p_1'{}^{\triangleright}$$
$$\Rightarrow a_2 \longrightarrow_{L_2} (p_1'{}^{\triangleright})^{\triangleright'}$$
$$\Rightarrow a_2 \longrightarrow_{L_2} p_1'{}^{\triangleright\triangleright'}$$

This completes the proof. □

Using abbreviations

$$L_1 =_{\text{def}} (\mathbf{P}(Loc \times \Sigma_L), \subseteq)$$
$$L_2 =_{\text{def}} (Loc \not\to (V \not\to \bigcup_{x \in X} L(D_x)), \sqsubseteq)$$

We will now define a GC $L_1 \underset{\triangleright}{\overset{\triangleleft}{\leftrightarrows}} L_2$ as follows:

(i) For $a \in L_1$, set

$$a^{\triangleright} =_{\text{def}} \delta$$
$$\text{dom } \delta =_{\text{def}} \pi_1(a)$$
$$\forall c \in \text{dom } \delta : \text{dom } \delta(c) =_{\text{def}} \bigcup_{\{\lambda \mid (c,\lambda) \in a\}} \text{dom } \lambda$$
$$\forall x \in \text{dom } \delta(c) : \delta(c)(x) =_{\text{def}} \bigsqcup \{\lambda(x) \mid (c, \lambda) \in a \wedge x \in \text{dom } \lambda\}$$

Observe that the last line of this definition can be equivalently replaced by

$$\forall c \in \text{dom } \delta : \delta(c) =_{\text{def}} \bigsqcup \{\lambda \mid (c, \lambda) \in a\}$$

(ii) For $\delta \in L_2$, set

$$\delta^{\triangleleft} =_{\text{def}} \{(c, \lambda) \mid c \in \text{dom } \delta \wedge \lambda \sqsubseteq \delta(c)\}$$

Lemma 1.11 *$\_^{\triangleright} : L_1 \to L_2$ and $\_^{\triangleleft} : L_2 \to L_1$ define a Galois connection.*

Proof. 1. Suppose $a^{\triangleright} \sqsubseteq \delta$. We show that this implies $a \subseteq \delta^{\triangleleft}$ by calculating

$$\begin{array}{l}
a^{\triangleright} \sqsubseteq \delta \Rightarrow \\
\quad \text{dom } a^{\triangleright} \subseteq \text{ dom } \delta \wedge (\forall c \in \text{dom } a^{\triangleright} : (a^{\triangleright})(c) \sqsubseteq \delta(c)) \Rightarrow \\
\quad \pi_1(a) \subseteq \text{dom } \delta \wedge (\forall c \in \pi_1(a) : \bigcup_{\{\lambda \mid (c,\lambda) \in a\}} \text{dom } \lambda \subseteq \text{dom } \delta(c)) \wedge \\
\qquad (\forall x \in \bigcup_{\{\lambda \mid (c,\lambda) \in a\}} \text{dom } \lambda : \bigsqcup \{\lambda(x) \mid (c, \lambda) \in a \wedge x \in \text{dom } \lambda\} \sqsubseteq \delta(c)(x)) \Rightarrow \\
\quad \delta^{\triangleleft} = \{(c, \lambda) \mid c \in \text{dom } \delta \wedge \lambda \sqsubseteq \delta(c)\} \\
\qquad \supseteq \{(c, \lambda) \mid c \in \pi_1(a) \wedge \lambda \sqsubseteq \delta(c)\} \\
\qquad = \{(c, \lambda) \mid c \in \pi_1(a) \wedge \text{dom } \lambda \subseteq \text{dom } \delta(c) \wedge (\forall x \in \text{dom } \lambda : \lambda(x) \sqsubseteq \delta(c)(x))\} \\
\qquad \supseteq \{(c, \lambda) \mid c \in \pi_1(a) \wedge \text{dom } \lambda \subseteq \bigcup_{\{\lambda' \mid (c,\lambda') \in a\}} \text{dom } \lambda' \wedge \\
\qquad\quad (\forall x \in \text{dom } \lambda : \lambda(x) \sqsubseteq \bigsqcup \{\lambda'(x) \mid (c, \lambda') \in a \wedge x \in \text{dom } \lambda'\})\} \\
\qquad \supseteq a
\end{array}$$

2. Suppose $a \subseteq \delta^{\triangleleft}$. We show that this implies $a^{\triangleright} \sqsubseteq \delta$ by calculating

$$\begin{array}{l} a \subseteq \delta^{\triangleleft} \Rightarrow \\ \quad a \subseteq \{(c,\lambda) \mid c \in \mathrm{dom}\ \delta \wedge \lambda \sqsubseteq \delta(c)\} \Rightarrow \\ \quad \forall c \in \pi_1(a), x \in \mathrm{dom}\ \delta(c) : \bigsqcup\{\lambda'(x) \mid (c,\lambda') \in a\} \sqsubseteq \delta(c)(x) \Rightarrow \\ \quad a^{\triangleright} \sqsubseteq \delta \end{array}$$

This completes the proof. □

Exploiting Theorem 1.10, we introduce a valid abstract interpretation on $L_2 = (Loc \rightharpoonup (V \rightharpoonup \bigcup_{x \in X}), \sqsubseteq)$ by means of the rules

$$\frac{p^{\triangleright\triangleleft} \longrightarrow_{L_1} p'}{p^{\triangleright} \longrightarrow_{L_2} p'^{\triangleright}}, \quad \frac{a_1{}^{\triangleleft} \longrightarrow_{L_1} p'}{a_1 \longrightarrow_{L_2} p'^{\triangleright}}$$

2 Abstract interpretation of while languages

The introductory results on valid abstract interpretations of transition systems of Section 1 will now be applied to a concrete programming language. We start with a simple C-like while-language, called G_1, initially only admitting main programs with global and stack variables, including arrays. Function calls will be considered at a later stage.

In the following exposition we require all Galois connections

$$\mathbf{P}(t) \underset{\triangleright}{\overset{\triangleleft}{\leftrightarrows}} L(t)$$

abstracting datatypes t to be exact in the sense that

$$\forall a \in L(t) : a^{\triangleleft\triangleright} = a$$

2.1 Preliminaries on array abstraction

We consider n-dimensional array variables a declared as

$$\mathtt{t\ a[d_1]} \ldots \mathtt{[d_n]};$$

as variables of partial n-ary function type, that is,

$$\mathtt{a} : \mathtt{int}^n \rightharpoonup \mathtt{t};\ (i_1, \ldots, i_n) \mapsto \mathtt{a[i_1]} \ldots \mathtt{[i_n]}$$

By choosing a lattice $L(\mathtt{int})$ abstracting array indexes and a lattice $L(t)$ abstracting the array type we wish to introduce an appropriate abstraction $\mathtt{a_L} : C(L(\mathtt{int})^n \rightharpoonup L(t))$ of array $\mathtt{a}$, where $C(L(\mathtt{int})^n \rightharpoonup L(t))$ is the set of continuous partial functions between lattices $L(\mathtt{int})^n$ and $L(t)$. To this end, it is necessary to introduce a GC

$$\mathbf{P}(\mathtt{int}^n \rightharpoonup \mathtt{t}) \underset{\triangleright}{\overset{\triangleleft}{\leftrightarrows}} C(L(\mathtt{int})^n \rightharpoonup L(t))$$

between the power set lattice $\mathbf{P}(\mathtt{int}^n \rightharpoonup \mathtt{t})$ and our lattice of interest $C(L(\mathtt{int})^n \rightharpoonup L(t))$. We define for some set $f_{\mathbf{P}} \in \mathbf{P}(\mathtt{int}^n \rightharpoonup \mathtt{t})$

$$\begin{array}{l} f_{\mathbf{P}}{}^{\triangleright} : C(L(\mathtt{int})^n \rightharpoonup L(t)); \\ (i_1^L, \ldots, i_n^L) \mapsto \bigsqcup\{\{f(i_1, \ldots, i_n)\}^{\triangleright} \mid f \in f_{\mathbf{P}} \wedge \forall j \in \{1, \ldots, n\} : i_j \in i_j^{L\triangleleft}\} \end{array}$$

Obviously, ${f_{\mathbf{P}}}^{\triangleright}$ is continuous, so $\_^{\triangleright}$ is well-defined. For $f_L : C(L(\mathtt{int})^n \not\to L(t))$ we define

$$
\begin{aligned}
{f_L}^{\triangleleft} = \{ f : \mathtt{int}^n \not\to \mathtt{t} \mid \mathrm{dom}\ f \subseteq (\mathrm{dom}\ f_L)^{\triangleleft} \wedge \\
\forall (i_1, \ldots, i_n) \in \mathrm{dom}\ f : \{f(i_1, \ldots, i_n)\}^{\triangleright} \sqsubseteq f_L(\{(i_1, \ldots, i_n)\}^{\triangleright})\}
\end{aligned}
$$

Lemma 2.1 $\mathbf{P}(\mathtt{int}^n \not\to \mathtt{t}) \underset{\triangleright}{\overset{\triangleleft}{\leftrightarrows}} C(L(\mathtt{int})^n \not\to L(t))$ *is a Galois Connection.*

Proof. Given existing GCs $\mathbf{P}(\mathtt{int}) \underset{\triangleright}{\overset{\triangleleft}{\leftrightarrows}} L(\mathtt{int})$ and $\mathbf{P}(t) \underset{\triangleright}{\overset{\triangleleft}{\leftrightarrows}} L(t)$, $f_{\mathbf{P}} \in \mathbf{P}(\mathtt{int}^n \not\to \mathtt{t})$ and $f_L : C(L(\mathtt{int})^n \not\to L(t))$ we first show

(a) $\mathbf{f_P \subseteq {f_L}^{\triangleleft} \Rightarrow {f_P}^{\triangleright} \sqsubseteq f_L}$.
${f_{\mathbf{P}}} \subseteq {f_L}^{\triangleleft}$ implies

$$
\begin{aligned}
\forall f \in f_{\mathbf{P}} : \mathrm{dom}\ f \subseteq (\mathrm{dom}\ f_L)^{\triangleleft} \wedge \\
\forall (i_1, \ldots, i_n) \in \mathrm{dom}\ f : \{f(i_1, \ldots, i_n)\}^{\triangleright} \sqsubseteq f_L(\{(i_1, \ldots, i_n)\}^{\triangleright})
\end{aligned}
$$

Therefore we can calculate

$$
\begin{aligned}
&\mathrm{dom}\ ({f_{\mathbf{P}}})^{\triangleright} \subseteq \mathrm{dom}\ f_L \\
&\forall (i_1^L, \ldots, i_n^L) \in \mathrm{dom}\ ({f_{\mathbf{P}}})^{\triangleright} : \\
&\quad ({f_{\mathbf{P}}})^{\triangleright}(i_1^L, \ldots, i_n^L) = \\
&\quad \bigsqcup \{ \{f(i_1, \ldots, i_n)\}^{\triangleright} \mid f \in f_{\mathbf{P}} \wedge \{(i_1, \ldots, i_n)\} \subseteq (i_1^L, \ldots, i_n^L)^{\triangleleft} \} \sqsubseteq \\
&\quad \bigsqcup \{ f_L(\{(i_1, \ldots, i_n)\}^{\triangleright}) \mid \{(i_1, \ldots, i_n)\}^{\triangleright} \sqsubseteq (i_1^L, \ldots, i_n^L) \} \sqsubseteq \\
&\quad f_L(i_1^L, \ldots, i_n^L) \quad \text{since } f_L \text{ is monotone}
\end{aligned}
$$

This proves ${f_{\mathbf{P}}}^{\triangleright} \sqsubseteq f_L$.

(b) $\mathbf{{f_P}^{\triangleright} \sqsubseteq f_L \Rightarrow f_P \subseteq {f_L}^{\triangleleft}}$.
${f_{\mathbf{P}}}^{\triangleright} \sqsubseteq f_L$ implies

$$
\begin{aligned}
&\mathrm{dom}\ ({f_{\mathbf{P}}})^{\triangleright} \subseteq \mathrm{dom}\ f_L \wedge \\
&\forall (i_1^L, \ldots, i_n^L) \in \mathrm{dom}\ ({f_{\mathbf{P}}})^{\triangleright} : ({f_{\mathbf{P}}})^{\triangleright}(i_1^L, \ldots, i_n^L) \sqsubseteq f_L(i_1^L, \ldots, i_n^L)
\end{aligned}
$$

From the definition of $\_^{\triangleright}$ we conclude

$$
\begin{aligned}
&\forall (i_1^L, \ldots, i_n^L) \in \mathrm{dom}\ ({f_{\mathbf{P}}})^{\triangleright} : \\
&\quad \bigsqcup \{ \{f(i_1, \ldots, i_n)\}^{\triangleright} \mid f \in f_{\mathbf{P}} \wedge \{(i_1, \ldots, i_n)\} \subseteq (i_1^L, \ldots, i_n^L)^{\triangleleft} \} \\
&\quad \sqsubseteq f_L(i_1^L, \ldots, i_n^L)
\end{aligned}
$$

For $(i_1^L, \ldots, i_n^L) =_{\mathrm{def}} \{(i_1, \ldots, i_n)\}^{\triangleright}$, this implies

$$
\forall f \in f_{\mathbf{P}}, \forall (i_1, \ldots, i_n) \in \mathrm{dom}\ f : \{f(i_1, \ldots, i_n)\}^{\triangleright} \sqsubseteq f_L(\{(i_1, \ldots, i_n)\}^{\triangleright})
$$

Now by definition of $\_^{\triangleleft}$,

$$f_L{}^{\triangleleft} = \{f : \mathtt{int}^n \not\to \mathtt{t} \mid \mathrm{dom}\ f \subseteq (\mathrm{dom}\ f_L)^{\triangleleft} \wedge \\ \forall (i_1,\ldots,i_n) \in \mathrm{dom}\ f : \{f(i_1,\ldots,i_n)\}^{\triangleright} \sqsubseteq f_L(\{(i_1,\ldots,i_n)\}^{\triangleright})\}$$

This implies $f_{\mathbf{P}} \subseteq f_L{}^{\triangleleft}$ and completes the proof. □

Lemma 2.2 *The GC* $\mathbf{P}(\mathtt{int}^n \not\to \mathtt{t}) \underset{\triangleright}{\overset{\triangleleft}{\leftrightarrows}} C(L(\mathtt{int})^n \not\to L(t))$ *is exact, that is,*

$$\forall f_L \in C(L(\mathtt{int})^n \not\to L(t)) : f_L{}^{\triangleleft\triangleright} = f_L$$

Proof. Given $f_L : C(L(\mathtt{int})^n \not\to L(t))$ and $(i_1^L,\ldots,i_n^L) \in \mathrm{dom} f_L$, we calculate

$$\begin{aligned} &f_L{}^{\triangleleft\triangleright}(i_1^L,\ldots,i_n^L) = \\ &\bigsqcup\{\{f(i_1,\ldots,i_n)\}^{\triangleright} \mid f \in f_L{}^{\triangleleft} \wedge \forall j \in \{1,\ldots,n\} : i_j \in i_j^{L\triangleleft}\} = \\ &\bigsqcup\{\{f(i_1,\ldots,i_n)\}^{\triangleright} \mid (\forall j \in \{1,\ldots,n\} : i_j \in i_j^{L\triangleleft}) \wedge \\ &\qquad \{f(i_1,\ldots,i_n)\}^{\triangleright} \sqsubseteq f_L(\{i_1\}^{\triangleright},\ldots,\{i_n\}^{\triangleright})\} = \\ &\bigsqcup\{f_L(\{i_1\}^{\triangleright},\ldots,\{i_n\}^{\triangleright}) \mid (\forall j \in \{1,\ldots,n\} : i_j \in i_j^{L\triangleleft})\} \end{aligned}$$

Since f_L is continuous, the last line of this equation can be re-written as

$$f_L{}^{\triangleleft\triangleright}(i_1^L,\ldots,i_n^L) = f_L(\bigsqcup\{\{i_1\}^{\triangleright} \mid i_1 \in i_1^{L\triangleleft}\},\ldots,\bigsqcup\{\{i_n\}^{\triangleright} \mid i_n \in i_n^{L\triangleleft}\})(*)$$

Since the GC $\mathbf{P}(\mathtt{int}) \underset{\triangleright}{\overset{\triangleleft}{\leftrightarrows}} L(\mathtt{int})$ is supposed to be exact, we can calculate for $j \in \{1,\ldots,n\}$:

$$\begin{aligned} i_j^L &= i_j^{L\triangleleft\triangleright} \\ &= \{i \mid i \in i_j^{L\triangleleft}\}^{\triangleright} \\ &= (\bigsqcup\{\{i\} \mid i \in i_j^{L\triangleleft}\})^{\triangleright} \\ &= \bigsqcup\{\{i\} \mid i \in i_j^{L\triangleleft}\}^{\triangleright} \\ &= \bigsqcup\{\{i\}^{\triangleright} \mid i \in i_j^{L\triangleleft}\} \end{aligned}$$

As a consequence, equation (*) can be simplified to

$$f_L{}^{\triangleleft\triangleright}(i_1^L,\ldots,i_n^L) = f_L(i_1^L,\ldots,i_n^L)$$

and this completes the proof. □

```
<G1>           ::= <global-defs>_opt <main>

<global-defs>  ::= <global-defs> <typedef> <id> <dimension>_opt;
                 | <typedef> <id> <dimension>_opt;

<typedef>      ::= int | bool | float | char

<id>           ::= C-Variable-Identifier

<dimension>    ::= <dimension>_opt [<nat>]

<nat>          ::= Non-zero natural number

<main>         ::= void main() { <body> }

<body>         ::= <stack-defs>_opt <commands>

<stack-defs>   ::= <stack-defs>_opt <typedef> <id>;

<commands>     ::= <commands>_opt <command>

<command>      ::= E
                 | ERROR
                 | ;
                 | <assignment>;
                 | if (<bexpr>) { <commands> } else { <commands> }
                 | while (<bexpr>) { <commands> }

<bexpr>        ::= <id_const>
                 | <id_const> <bop> <id_const>
                 | not <id_const>

<bop>          ::= < | <= | > | >= | and | or

<assignment>   ::= <id> = <rhs>
                 | <id> = <id>[<int>]
                 | <id>[<int>] = <id_const>

<int>          ::= Integral number

<rhs>          ::= <id_const> <op> <id_const>
                 | <id_const>
                 | - <id_const>
                 | input()
                 | (<typedef>) <id_const>

<op>           ::= + | - | * | /

<id_const>     ::= <id> | <const>

<const>        ::= Literal defining a constant value
```

Fig. 1. Syntax of the simple while-language G_1.

(**D1**) $\dfrac{x \text{ is a global variable of type } \mathtt{t}}{\langle \mathtt{t\ x;B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B},\sigma[\mathtt{x}\mapsto 0]\rangle}$

(**D2**) $\dfrac{x \text{ is a local variable of type } \mathtt{t}}{\langle \mathtt{t\ x;B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B},\sigma[\mathtt{x}\mapsto ?]\rangle}$

(**D3**) $\dfrac{x \text{ is a global array of type } \mathtt{t} \wedge n{>}0}{\langle \mathtt{t\ x[n];B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B},\sigma[\mathtt{x[i]}\mapsto 0 \mid i{\in}\{0,\ldots,n{-}1\}]\rangle}$

(**D4**) $\dfrac{x \text{ is a local array of type } \mathtt{t} \wedge n{>}0}{\langle \mathtt{t\ x;B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B},\sigma[\mathtt{x[i]}\mapsto ? \mid i{\in}\{0,\ldots,n{-}1\}]\rangle}$

(**AS1**) $\dfrac{\sigma(\mathtt{x_2}){\neq}?}{\langle \mathtt{x_1{=}x_2;B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B},\sigma[\mathtt{x_1}\mapsto\sigma(\mathtt{x_2})]\rangle}$

(**AS2**) $\dfrac{\sigma(\mathtt{x_2}){\neq}?}{\langle \mathtt{x_1{=}{-}x_2;B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B},\sigma[\mathtt{x_1}\mapsto -\sigma(\mathtt{x_2})]\rangle}$

(**AS3**) $\dfrac{\omega{\in}\{+,-,*\}\wedge\sigma(x_2){\neq}?\wedge\sigma(x_3){\neq}?}{\langle \mathtt{x_1{=}x_2}\omega\mathtt{x_3;B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B},\sigma[\mathtt{x_1}\mapsto\sigma(\mathtt{x_2})\omega\sigma(\mathtt{x_3})]\rangle}$

(**AS4**) $\dfrac{\sigma(x_2){\neq}?\wedge\sigma(x_3){\notin}\{0,?\}}{\langle \mathtt{x_1{=}x_2/x_3;B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B},\sigma[\mathtt{x_1}\mapsto\sigma(\mathtt{x_2})/\sigma(\mathtt{x_3})]\rangle}$

(**AS5**) $\dfrac{}{\langle \mathtt{x_1{=}input();B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B},\sigma[x_1\mapsto\sigma(\mathtt{input()})]\rangle}$

(**AS6**) $\dfrac{\sigma(i){\in}\{0,\ldots,\mathtt{dim(x_2)}{-}1\}\wedge\sigma(\mathtt{x_2[\sigma(i)]}){\neq}?}{\langle \mathtt{x_1{=}x_2[i];B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B},\sigma[\mathtt{x_1}\mapsto\sigma(\mathtt{x_2[\sigma(i)]})]\rangle}$

(**AS7**) $\dfrac{\sigma(i){\in}\{0,\ldots,\mathtt{dim(x_1)}{-}1\}\wedge\sigma(\mathtt{x_2}){\neq}?}{\langle \mathtt{x_1[i]{=}x_2;B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B},\sigma[\mathtt{x_1[\sigma(i)]}\mapsto\sigma(\mathtt{x_2})]\rangle}$

(**E1**) $\dfrac{}{\langle \mathtt{E},\sigma\rangle \not\longrightarrow_{W_1}}$

(**SC1**) $\dfrac{}{\langle \mathtt{;B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B},\sigma\rangle}$

(**SC2**) $\dfrac{\langle \mathtt{B_1},\sigma_1\rangle \longrightarrow_{W_1} \langle \mathtt{E},\sigma_2\rangle}{\langle \mathtt{B_1;B_2},\sigma_1\rangle \longrightarrow_{W_1} \langle \mathtt{B_2},\sigma_2\rangle}$

(**IF1**) $\dfrac{(\forall i{\in}\{1,\ldots,n\}{:}\sigma(x_i){\neq}?){\wedge}\mathtt{b}(\sigma(\mathtt{x_1}),\ldots,\sigma(\mathtt{x_n}))}{\langle \mathtt{if(b(x_1,\ldots,x_n))\{B_1\}else\{B_2\}},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B_1},\sigma\rangle}$

(**IF2**) $\dfrac{(\forall \mathtt{i}{\in}\{1,\ldots,\mathtt{n}\}{:}\sigma(\mathtt{x_i}){\neq}?){\wedge}\neg\mathtt{b}(\sigma(\mathtt{x_1}),\ldots,\sigma(\mathtt{x_n}))}{\langle \mathtt{if(b(x_1,\ldots,x_n))\{B_1\}else\{B_2\}},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B_2},\sigma\rangle}$

(**WH1**) $\dfrac{(\forall i{\in}\{1,\ldots,n\}{:}\sigma(x_i){\neq}?){\wedge}\mathtt{b}(\sigma(\mathtt{x_1}),\ldots,\sigma(\mathtt{x_n}))}{\langle \mathtt{while(b(x_1,\ldots,x_n))\{B_1\};B_2},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B_1;while(b(x_1,\ldots,x_n))\{B_1\};B_2},\sigma\rangle}$

(**WH2**) $\dfrac{(\forall \mathtt{i}{\in}\{1,\ldots,\mathtt{n}\}{:}\sigma(\mathtt{x_i}){\neq}?){\wedge}\neg\mathtt{b}(\sigma(\mathtt{x_1}),\ldots,\sigma(\mathtt{x_n}))}{\langle \mathtt{while(b(x_1,\ldots,x_n))\{B_1\};B_2},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{B_2},\sigma\rangle}$

Fig. 2. Operational semantics of the simple while-language G_1 – well-defined state transitions.

(**ER1**) $\frac{\sigma(x_3)\in\{0,?\}}{\langle \mathtt{x_1=x_2/x_3;B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{ERROR},?\rangle}$

(**ER2**) $\frac{\sigma(\mathtt{x_2})=?}{\langle \mathtt{x_1=e(\ldots,x_2,\ldots);},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{ERROR},?\rangle}$

(**ER3**) $\frac{\sigma(i)\notin\{0,\ldots,\mathtt{dim(x_2)}-1\}\vee\sigma(\mathtt{x_2[i]})=?}{\langle \mathtt{x_1=x_2[i];B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{ERROR},?\rangle}$

(**ER4**) $\frac{\sigma(i)\notin\{0,\ldots,\mathtt{dim(x_1)}-1\}\vee\sigma(\mathtt{x_2})=?}{\langle \mathtt{x_1[i]=x_2;B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{ERROR},?\rangle}$

(**ER5**) $\frac{}{\langle \mathtt{ERROR},\sigma\rangle \not\longrightarrow_{W_1}}$

(**ER6**) $\frac{(\exists i\in\{1,\ldots,n\}:\sigma(x_i)=?)}{\langle \mathtt{if(b(x_1,\ldots,x_n))\{B_1\}else\{B_2\}},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{ERROR},?\rangle}$

(**ER7**) $\frac{(\exists i\in\{1,\ldots,n\}:\sigma(x_i)=?)}{\langle \mathtt{while\ (b(x_1,\ldots,x_n))\ B},\sigma\rangle \longrightarrow_{W_1} \langle \mathtt{ERROR},?\rangle}$

Fig. 3. Operational semantics of the simple while-language G_1 – runtime errors.

2.2 Abstract Semantics of Coarse-Grained Abstraction L_2

Applying Lemma 1.2 tells us how to lift the G_1-operations $\omega \in \{+,-,*,/\}$ to the lattices L chosen for data abstraction:

$$a\omega_L b = \bigsqcup\{\{x\omega y\}^{\triangleright} \mid x \in a^{\triangleleft} \wedge y \in b^{\triangleleft}\}$$

(**AD1**) $\dfrac{x \text{ is a global variable of type } t}{\{(\mathtt{t\ x;B}) \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \delta[\mathtt{x} \mapsto \{0\}^{\triangleright}]\}}$

(**AD2**) $\dfrac{x \text{ is a local variable of type } t}{\{(\mathtt{t\ x;B}) \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \delta[\mathtt{x} \mapsto \bot]\}}$

(**AD3**) $\dfrac{x \text{ is a global array of type } t \wedge n{>}0}{\{(\mathtt{t\ x[n];B}) \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \delta[\mathtt{x[i]} \mapsto \{0\}^{\triangleright} \mid i \in \{0,\ldots,n-1\}]\}}$

(**AD4**) $\dfrac{x \text{ is a local array of type } t \wedge n{>}0}{\{(\mathtt{t\ x;B}) \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \delta[\mathtt{x[i]} \mapsto \bot \mid i \in \{0,\ldots,n-1\}]\}}$

(**AAS1**) $\dfrac{\delta(\mathtt{x_2}) \neq \bot}{\{(\mathtt{x_1{=}x_2;B}) \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \delta[\mathtt{x_1} \mapsto \delta(\mathtt{x_2})]\}}$

(**AAS2**) $\dfrac{\delta(\mathtt{x_2}) \neq \bot}{\{\mathtt{x_1{=}{-}x_2;B} \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \delta[\mathtt{x_1} \mapsto (-_{L_2})\delta(\mathtt{x_2})]\}}$

(**ASC1**) $\dfrac{}{\{(\mathtt{;B}) \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \delta\}}$

(**AAS5**) $\dfrac{}{\{(\mathtt{x_1{=}input();B}) \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \delta[\mathtt{x_1} \mapsto \delta(\mathtt{input()})]\}}$

(**AAS6**) $\dfrac{\delta(i)^{\triangleleft} \subseteq \{0,\ldots,\mathtt{dim(x_2)}-1\} \wedge \delta(\mathtt{x_2}[\delta(\mathtt{i})]) \neq \bot}{\{\mathtt{x_1{=}x_2[i];B} \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \delta[\mathtt{x_1} \mapsto \delta(\mathtt{x_2}[\delta(\mathtt{i})])]\}}$

(**AAS7**) $\dfrac{\delta(i)^{\triangleleft} \subseteq \{0,\ldots,\mathtt{dim(x_1)}-1\} \wedge \delta(\mathtt{x_2}) \neq \bot}{\{\mathtt{x_1[i]{=}x_2;B} \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \delta[\mathtt{x_1} \mapsto a']\}}$
$a'[k_L] =_{\text{def}} \{\delta(x_1)[k_L] \mid (\#\delta(i)^{\triangleleft} \geq 2 \vee k_L \sqcap \delta(i) = \bot)\} \sqcup \{\delta(x_2) \mid k_L \sqcap \delta(i) \neq \bot\}$

(**AE1**) $\dfrac{}{\{\mathtt{E} \mapsto \delta\} \not\longrightarrow_{L_2}}$

(**ASC2**) $\dfrac{\{\mathtt{B_1} \mapsto \delta_1\} \longrightarrow_{L_2} \{\mathtt{E} \mapsto \delta_2\}}{\{\mathtt{B_1;B_2} \mapsto \delta_1\} \longrightarrow_{L_2} \{\mathtt{B_2} \mapsto \delta_2\}}$

(**AAS3**) $\dfrac{\mathtt{Type(x_1){=}Type(x_2){=}Type(x_3){=}t} \in \{\mathtt{int,float}\} \wedge \omega \in \{+,-,*\} \wedge \delta(x_2) \neq \bot \wedge \delta(x_3) \neq \bot}{\{(\mathtt{x_1{=}x_2}\,\omega\,\mathtt{x_3;B}) \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \delta[\mathtt{x_1} \mapsto \delta(\mathtt{x_2})(\omega_{L(t)})\delta(\mathtt{x_3})]\}}$

(**AAS4**) $\dfrac{\mathtt{Type(x_1){=}Type(x_2){=}Type(x_3){=}t} \in \{\mathtt{int,float}\} \wedge \delta(x_2) \neq \bot \wedge \delta(x_3) \notin \{\{0\}^{\triangleright}, \bot\}}{\{\mathtt{x_1{=}x_2/x_3;B}) \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \delta[\mathtt{x_1} \mapsto \delta(\mathtt{x_2})(/_{L(t)})\delta(\mathtt{x_3})]\}}$

(**AIF1**) $\dfrac{(\forall i \in \{1,\ldots,n\} : \delta(x_i) \neq \bot) \wedge \mathtt{b}_{\mathtt{L(bool)}}(\delta(\mathtt{x_1}),\ldots,\delta(\mathtt{x_n})) = \mathtt{true}}{\{\mathtt{if(b(x_1,\ldots,x_n))\{B_1\}else\{B_2\}} \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B_1} \mapsto \delta\}}$

(**AIF2**) $\dfrac{(\forall \mathtt{i} \in \{1,\ldots,n\} : \delta(\mathtt{x_i}) \neq \bot) \wedge \mathtt{b}_{\mathtt{L(bool)}}(\delta(\mathtt{x_1}),\ldots,\delta(\mathtt{x_n})) = \mathtt{false}}{\{\mathtt{if(b(x_1,\ldots,x_n))\{B_1\}else\{B_2\}} \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B_2} \mapsto \delta\}}$

(**AIF3**) $\dfrac{(\forall i \in \{1,\ldots,n\} : \delta(x_i) \neq \bot) \wedge \mathtt{b}_{\mathtt{L(bool)}}(\delta(\mathtt{x_1}),\ldots,\delta(\mathtt{x_n})) = \top}{\{\mathtt{if(b(x_1,\ldots,x_n))\{B_1\}else\{B_2\}} \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B_1} \mapsto \delta[x_i \mapsto \pi_i(((\delta(x_1)^{\triangleleft} \times \ldots \times \delta(x_n)^{\triangleleft}) \cap b^{-1}(\{\mathtt{true}\}))^{\triangleright}) \mid i \in \{1,\ldots,n\}], \mathtt{B_2} \mapsto \delta[x_i \mapsto \pi_i(((\delta(x_1)^{\triangleleft} \times \ldots \times \delta(x_n)^{\triangleleft}) \cap b^{-1}(\{\mathtt{false}\}))^{\triangleright}) \mid i \in \{1,\ldots,n\}]\}}$

(**AWH1**) $\dfrac{(\forall i \in \{1,\ldots,n\} : \delta(x_i) \neq \bot) \wedge \mathtt{b}_{\mathtt{L(bool)}}(\delta(\mathtt{x_1}),\ldots,\delta(\mathtt{x_n})) = \mathtt{true}}{\{\mathtt{while(b(x_1,\ldots,x_n))\{B_1\};B_2} \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B_1;while(b(x_1,\ldots,x_n))\{B_1\};B_2} \mapsto \delta\}}$

(**AWH2**) $\dfrac{(\forall \mathtt{i} \in \{1,\ldots,n\} : \delta(\mathtt{x_i}) \neq \bot) \wedge \mathtt{b}_{\mathtt{L(bool)}}(\delta(\mathtt{x_1}),\ldots,\delta(\mathtt{x_n})) = \mathtt{false}}{\{\mathtt{while(b(x_1,\ldots,x_n))\{B_1\};B_2} \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B_2} \mapsto \delta\}}$

(**AWH3**) $\dfrac{(\forall \mathtt{i} \in \{1,\ldots,n\} : \delta(\mathtt{x_i}) \neq \bot) \wedge \mathtt{b}_{\mathtt{L(bool)}}(\delta(\mathtt{x_1}),\ldots,\delta(\mathtt{x_n})) = \top}{\{\mathtt{while(b(x_1,\ldots,x_n))\{B_1\};B_2} \mapsto \delta\} \longrightarrow_{L_2} \{\mathtt{B_1;while(b(x_1,\ldots,x_n))\{B_1\};B_2} \mapsto \delta[x_i \mapsto \pi_i(((\delta(x_1)^{\triangleleft} \times \ldots \times \delta(x_n)^{\triangleleft}) \cap b^{-1}(\{\mathtt{true}\}))^{\triangleright}) \mid i \in \{1,\ldots,n\}], \mathtt{B_2} \mapsto \delta[x_i \mapsto \pi_i(((\delta(x_1)^{\triangleleft} \times \ldots \times \delta(x_n)^{\triangleleft}) \cap b^{-1}(\{\mathtt{false}\}))^{\triangleright}) \mid i \in \{1,\ldots,n\}]\}}$

(**AU1**) $\dfrac{\lambda_1 \longrightarrow_{L_2} \lambda_1' \wedge \lambda_2 \longrightarrow_{L_2} \lambda_2' \wedge \text{dom } \lambda_1 \cap \text{dom } \lambda_2 = \varnothing}{\lambda_1 \cup \lambda_2 \longrightarrow_{L_2} \lambda_1' \sqcup \lambda_2'}$

Fig. 4. Abstract semantics of the simple while-language G_1 (coarse-grained abstraction): The well-defined state transitions.

$$\frac{\delta(x_3)\in\{\{0\}^{\triangleright},\bot\}}{\{(\mathtt{x_1}=\mathtt{x_2}/\mathtt{x_3};\mathtt{B})\mapsto\delta\}\longrightarrow_{L_2}\{\mathtt{ERROR}\mapsto\bot\}}$$

$$\frac{\delta(\mathtt{x_2})=\bot}{\{(\mathtt{x_1}=\mathtt{e}(\ldots,\mathtt{x_2},\ldots);)\mapsto\delta\}\longrightarrow_{L_2}\{\mathtt{ERROR}\mapsto\bot\}}$$

$$\frac{\delta(i)\notin\{0,\ldots,\mathtt{dim}(\mathtt{x_2})-1\}\vee\delta(\mathtt{x_2}[\mathtt{i}])=\bot}{\{\mathtt{x_1}=\mathtt{x_2}[\mathtt{i}];\mathtt{B}\mapsto\delta\}\longrightarrow_{L_2}\{\mathtt{ERROR}\mapsto\bot\}}$$

$$\frac{\delta(i)\notin\{0,\ldots,\mathtt{dim}(\mathtt{x_1})-1\}\vee\delta(\mathtt{x_2})=\bot}{\{\mathtt{x_1}[\mathtt{i}]=\mathtt{x_2};\mathtt{B}\mapsto\delta\}\longrightarrow_{L_2}\{\mathtt{ERROR}\mapsto\bot\}}$$

$$\frac{}{\{\mathtt{E}\mapsto\delta\}\not\longrightarrow_{L_2}} \qquad \frac{}{\{\mathtt{ERROR}\mapsto\delta\}\not\longrightarrow_{L_2}}$$

$$\frac{(\exists i\in\{1,\ldots,n\}:\delta(x_i)=\bot)}{\{\mathtt{if(b(x_1,\ldots,x_n))\{B_1\}else\{B_2\}}\mapsto\delta\}\longrightarrow_{L_2}\{\mathtt{ERROR}\mapsto\bot\}} \qquad \frac{(\exists i\in\{1,\ldots,n\}:\delta(x_i)=\bot)}{\{\mathtt{while\ (b(x_1,\ldots,x_n))\ B}\mapsto\delta\}\longrightarrow_{L_2}\{\mathtt{ERROR}\mapsto\bot\}}$$

Fig. 5. Abstract semantics of the simple while-language G_1 (coarse-grained abstraction): State transitions leading to runtime errors.

Theorem 2.3 *The rules of Fig. 5 define a valid abstract interpretation for G_1 in the sense of Definition 1.6.*

Proof. 1. Validity of rules (AAS1),(AAS2),(AAS3),(AAS4),(AAS5). We prove the validity of (AAS3), the other proofs are constructed in an analogous way.

Step 1. We calculate the corresponding semantic rule for $L_1 = \mathbf{P}(Loc \times \Sigma_L)$, using the fact that the abstract interpretation semantics on L_1 has been introduced according to Theorem 1.7.

We calculate for a set $B \subseteq \Sigma_L$ using the definition of the GC between $\mathbf{P}(Loc\times\Sigma)$ and L_1:

$$\begin{aligned}&\{((\mathtt{x_1}=\mathtt{x_2}\omega\mathtt{x_3};\mathtt{B}),\delta)\mid\delta\in B\}^{\triangleleft}=\\&\quad\{((\mathtt{x_1}=\mathtt{x_2}\omega\mathtt{x_3};\mathtt{B}),\sigma)\mid\exists\delta\in B:\mathrm{dom}\,\sigma=\mathrm{dom}\,\delta\wedge(\forall x\in\mathrm{dom}\,\sigma:\{\sigma(x)\}\subseteq\delta(x)^{\triangleleft})\}\\&\quad\longrightarrow_{\mathbf{P}}p'\end{aligned}$$

with

$$\begin{aligned}p'=_{\mathrm{def}}\{(\mathtt{B},\sigma[x_1\mapsto\sigma(x_2)\omega\sigma(x_3)])\mid\exists\delta\in B:\mathrm{dom}\,\sigma=\mathrm{dom}\,\delta\wedge\\(\forall x\in\mathrm{dom}\,\sigma:\{\sigma(x)\}\subseteq\delta(x)^{\triangleleft})\}\end{aligned}$$

Since the transition relation $\longrightarrow_{L_1}$ has been introduced according to the rules of Theorem 1.7, we can apply rule (iii) of this theorem and conclude

$$\{((\mathtt{x_1}=\mathtt{x_2}\omega\mathtt{x_3};\mathtt{B}),\delta)\mid\delta\in B\}\longrightarrow_{L_1}p'^{\triangleright}$$

with $p'^{\triangleright}$ calculated according to the definition of $\_^{\triangleright}$ and by application of Lemma 1.8 as

$$\begin{aligned}p'^{\triangleright}=\{(\mathtt{B},\{y\mapsto\{\sigma(y)\}^{\triangleright}\mid y\in\mathrm{dom}\,\sigma-\{x_1\}\}\cup\{x_1\mapsto\{\sigma(x_2)\omega\sigma(x_3)\}^{\triangleright}\})\mid\\\exists\delta\in B:\mathrm{dom}\,\sigma=\mathrm{dom}\,\delta\wedge(\forall x\in\mathrm{dom}\,\sigma:\{\sigma(x)\}\subseteq\delta(x)^{\triangleleft})\}\end{aligned}$$

Step 2. We calculate the corresponding semantic rule for $L_2 = (Loc \nrightarrow \Sigma_L)$, using the fact that the abstract interpretation semantics on L_2 has been introduced according to Theorem 1.10 with respect to L_1. To this end, we calculate according to

the GC definitions on page 9 for $B =_{\text{def}} \{\delta \in \Sigma_L \mid \delta \sqsubseteq \lambda\}$:

$$\{(\mathtt{x_1} = \mathtt{x_2}\omega\mathtt{x_3}; \mathtt{B}) \mapsto \lambda\}^{\triangleleft} = \{((\mathtt{x_1} = \mathtt{x_2}\omega\mathtt{x_3}; \mathtt{B}), \delta) \mid \delta \in B\}$$

$$\{(\mathtt{x_1} = \mathtt{x_2}\omega\mathtt{x_3}; \mathtt{B}, \delta) \mid \delta \in B\} \longrightarrow_{L_1} q'$$

$$q' =_{\text{def}} \{(\mathtt{B}, \{y \mapsto \{\sigma(y)\}^{\triangleright} \mid y \in \text{dom } \sigma - \{x_1\}\} \cup \{x_1 \mapsto \{\sigma(x_2)\omega\sigma(x_3)\}^{\triangleright}\}) \mid$$
$$\exists \delta \in B : \text{dom } \sigma = \text{dom } \delta \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\}$$

From Theorem 1.10 we conclude

$$\{(\mathtt{x_1} = \mathtt{x_2}\omega\mathtt{x_3}; \mathtt{B}) \mapsto \lambda\} \longrightarrow_{L_2} {q'}^{\triangleright}$$

and calculate according to the GC definitions on page 9

$${q'}^{\triangleright} = \{\mathtt{B} \mapsto \lambda'\}$$

with

$$\text{dom } \lambda' = \text{dom } \lambda$$

$$\forall y \in \text{dom } \lambda' - \{x_1\} : \lambda'(y) =$$
$$\bigsqcup\{\{\sigma(y)\}^{\triangleright} \mid \exists \delta \in B : \text{dom } \sigma = \text{dom } \delta \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\} =$$
$$\bigsqcup\{\{\sigma(y)\}^{\triangleright} \mid \text{dom } \sigma \subseteq \text{dom } \lambda \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \lambda(x)^{\triangleleft})\} = \quad (1)$$
$$\bigsqcup(\{\{\sigma(y)\} \mid \text{dom } \sigma \subseteq \text{dom } \lambda \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \lambda(x)^{\triangleleft})\}^{\triangleright}) = \quad (2)$$
$$(\bigsqcup\{\{\sigma(y)\} \mid \text{dom } \sigma \subseteq \text{dom } \lambda \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \lambda(x)^{\triangleleft})\})^{\triangleright} = \quad (3)$$
$$\{\sigma(y) \mid \sigma(y) \in \lambda(y)^{\triangleleft}\}^{\triangleright} = \quad (4)$$
$$\lambda(y)^{\triangleleft\triangleright} = \quad (5)$$
$$\lambda(y)$$

$$\lambda'(x_1) = \bigsqcup\{\{\sigma(x_2)\omega\sigma(x_3)\}^{\triangleright} \mid$$
$$\exists \delta \in B : \text{dom } \sigma = \text{dom } \delta \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\} =$$
$$\bigsqcup\{\{\sigma(x_2)\omega\sigma(x_3)\}^{\triangleright} \mid \sigma(x_2) \in \lambda(x_2)^{\triangleleft} \wedge \sigma(x_3) \in \lambda(x_3)^{\triangleleft}\} =$$
$$\lambda(x_2)\omega_L\lambda(x_1)$$

In line (1) of the calculations above we have just re-written the function application of $\_^{\triangleright}$ according to $X^{\triangleright} =_{\text{def}} \{x^{\triangleright} \mid x \in X\}$. In line (2) we have used the fact that $\_^{\triangleright}$ preserves joins (i. e., $(\bigsqcup X)^{\triangleright} = \bigsqcup(X^{\triangleright})$). The equality (3) follows from the fact that for power set lattices, the supremum is just the union of sets. Finally, equality (5) follows from the exactness of the GCs defining the datatype abstraction for each variable symbol y. In the last line in the calculation of $\lambda'(x_1)$ we have used Lemma 1.2.

Combining the results elaborated for $\lambda'(y)$ and $\lambda'(x_1)$ results in

$$\{(\mathtt{x_1} = \mathtt{x_2}\omega\mathtt{x_3}; \mathtt{B}) \mapsto \lambda\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \lambda[x_1 \mapsto \lambda(x_2)\omega_L\lambda(x_1)]\}$$

so this proves (AAS3).

2. Validity of rule (AAS6). We proceed in an analogy to the proof for the rules in 1.

Step 1. We calculate the corresponding semantic rule for $L_1 = \mathbf{P}(Loc \times \Sigma_L)$ for a set $B \subseteq \Sigma_L$, using the definition of the GC between $\mathbf{P}(Loc \times \Sigma)$ and L_1:

$$\begin{aligned}&\{((\mathtt{x_1} = \mathtt{x_2[i]}; \mathtt{B}), \delta) \mid \delta \in B\}^{\triangleleft} = \\ &\quad \{((\mathtt{x_1} = \mathtt{x_2[i]}; \mathtt{B}), \sigma) \mid \exists \delta \in B : \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta \wedge (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\} \\ &\quad \longrightarrow_{\mathbf{P}} p'\end{aligned}$$

with

$$\begin{aligned}p' =_{\mathrm{def}} \{(\mathtt{B}, \sigma[x_1 \mapsto \sigma(x_2)[\sigma(i)]]) \mid \exists \delta \in B : \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta \wedge \\ (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\}\end{aligned}$$

Since the transition relation $\longrightarrow_{L_1}$ has been introduced according to the rules of Theorem 1.7, we can apply rule (iii) of this theorem and conclude

$$\{((\mathtt{x_1} = \mathtt{x_2[i]}; \mathtt{B}), \delta) \mid \delta \in B\} \longrightarrow_{L_1} p'^{\triangleright}$$

with $p'^{\triangleright}$ calculated according to the definition of $\_^{\triangleright}$ and by application of Lemma 1.8 as

$$\begin{aligned}p'^{\triangleright} = \{(\mathtt{B}, \{y \mapsto \{\sigma(y)\}^{\triangleright} \mid y \in \mathrm{dom}\ \sigma - \{x_1\}\} \cup \{x_1 \mapsto \{\sigma(x_2)[\sigma(i)]\}^{\triangleright}\}) \mid \\ \exists \delta \in B : \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta \wedge (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\}\end{aligned}$$

Step 2. We calculate the corresponding semantic rule for $L_2 = (Loc \nrightarrow \Sigma_L)$ as above in 1., Step 2, with $B =_{\mathrm{def}} \{\delta \in \Sigma_L \mid \delta \sqsubseteq \lambda\}$:

$$\begin{aligned}&\{(\mathtt{x_1} = \mathtt{x_2[i]}; \mathtt{B}) \mapsto \lambda\}^{\triangleleft} = \{((\mathtt{x_1} = \mathtt{x_2[i]}; \mathtt{B}), \delta) \mid \delta \in B\} \\ &\{(\mathtt{x_1} = \mathtt{x_2[i]}; \mathtt{B}, \delta) \mid \delta \in B\} \longrightarrow_{L_1} q' \\ &q' =_{\mathrm{def}} \{(\mathtt{B}, \{y \mapsto \{\sigma(y)\}^{\triangleright} \mid y \in \mathrm{dom}\ \sigma - \{x_1\}\} \cup \{x_1 \mapsto \{\sigma(x_2)[\sigma(i)]\}^{\triangleright} \mid \\ &\qquad \exists \delta \in B : \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta \wedge (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\}\end{aligned}$$

From Theorem 1.10 we conclude

$$\{(\mathtt{x_1} = \mathtt{x_2[i]}; \mathtt{B}) \mapsto \lambda\} \longrightarrow_{L_2} q'^{\triangleright}$$

and calculate according to the GC definitions on page 9

$$q'^{\triangleright} = \{\mathtt{B} \mapsto \lambda'\}$$

with dom $\lambda' = $ dom λ and $\forall y \in$ dom $\lambda' - \{x_1\} : \lambda'(y) = \lambda(y)$, as can be shown exactly as in 1. It remains to calculate the function value $\lambda'(x_1)$:

$$\begin{aligned}
\lambda'(x_1) = \bigsqcup\{\{\sigma(x_2)[\sigma(i)]\}^{\triangleright} \mid \\
\exists \delta \in B : \text{dom } \sigma = \text{dom } \delta \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\} = \\
\bigsqcup\{\{\sigma(x_2)[\sigma(i)]\}^{\triangleright} \mid \text{dom } \sigma \subseteq \text{dom } \lambda \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \lambda(x)^{\triangleleft})\} = \\
\bigsqcup\{\{\sigma(x_2)[\sigma(i)]\}^{\triangleright} \mid \sigma(x_2) \in \lambda(x_2)^{\triangleleft} \wedge \sigma(i) \in \lambda(i)^{\triangleleft}\} = \quad (1) \\
\lambda(x_2)^{\triangleleft\triangleright}[\lambda(i)] = \quad (2) \\
\lambda(x_2)[\lambda(i)]
\end{aligned}$$

Equality (1) in the calculation above follows from the GC definition for array variables given on page 10, equality (2) follows from the fact that this GC is exact (Lemma 2.2).

Combining the results elaborated for $\lambda'(y)$ and $\lambda'(x_1)$ results in

$$\{(\mathtt{x_1} = \mathtt{x_2}[\mathtt{i}]; \mathtt{B}) \mapsto \lambda\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \lambda[x_1 \mapsto \lambda(x_2)[\lambda(i)]]\}$$

so this proves (AAS6).

3. Validity of rules (AIF1),(AIF2),(AIF3). We prove the validity of (AIF3), the proofs for (AIF1), (AIF2) are constructed in an analogous way.

Step 1. Again, we first calculate the corresponding semantic rule for $L_1 = \mathbf{P}(Loc \times \Sigma_L)$. Setting

$$c =_{\text{def}} \mathtt{if(b(x_1, \ldots, x_n))\{B_1\}else\{B_2\}}$$

we calculate for a set $B \subseteq \Sigma_L$ using the definition of the GC between $\mathbf{P}(Loc \times \Sigma)$ and L_1:

$$\begin{aligned}
&\{(c, \delta) \mid \delta \in B\}^{\triangleleft} = \\
&\quad \{(c, \sigma) \mid \exists \delta \in B : \text{dom } \sigma = \text{dom } \delta \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\} \\
&\quad \longrightarrow_{\mathbf{P}} p'_1 \cup p'_2
\end{aligned}$$

with

$$\begin{aligned}
p'_1 =_{\text{def}} \{(\mathtt{B_1}, \sigma) \mid \exists \delta \in B : \text{dom } \sigma = \text{dom } \delta \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft}) \wedge \\
\mathtt{b}(\sigma(\mathtt{x_1}), \ldots, \sigma(\mathtt{x_n})) = \mathtt{true}\} \\
p'_2 =_{\text{def}} \{(\mathtt{B_2}, \sigma) \mid \exists \delta \in B : \text{dom } \sigma = \text{dom } \delta \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft}) \wedge \\
\mathtt{b}(\sigma(\mathtt{x_1}), \ldots, \sigma(\mathtt{x_n})) = \mathtt{false}\}
\end{aligned}$$

Applying rule (iii) of Theorem 1.7 and using the definition of $\_^{\triangleright}$ on page 7 yields

$$
\begin{aligned}
&\{(c,\delta) \mid \delta \in B\} \longrightarrow_{L_1} (p_1' \cup p_2')^{\triangleright} \\
&(p_1' \cup p_2')^{\triangleright} = \\
&\quad \{(\mathtt{B_1},\delta') \mid \exists \sigma \in \Sigma : (\mathtt{B_1},\sigma) \in p_1' \wedge \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta' \wedge \\
&\qquad\qquad (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\}^{\triangleright} = \delta'(x))\} \cup \\
&\quad \{(\mathtt{B_2},\delta') \mid \exists \sigma \in \Sigma : (\mathtt{B_2},\sigma) \in p_2' \wedge \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta' \wedge \\
&\qquad\qquad (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\}^{\triangleright} = \delta'(x))\} = \\
&\quad \{(\mathtt{B_1},\delta') \mid \exists \sigma \in \Sigma : \exists \delta \in B : \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta = \mathrm{dom}\ \delta' \wedge \\
&\qquad\qquad (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft} \wedge \{\sigma(x)\}^{\triangleright} = \delta'(x)) \wedge \\
&\qquad\qquad b(\sigma(x_1),\ldots,\sigma(x_n)) = \mathtt{true}\} \cup \\
&\quad \{(\mathtt{B_2},\delta') \mid \exists \sigma \in \Sigma : \exists \delta \in B : \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta = \mathrm{dom}\ \delta' \wedge \\
&\qquad\qquad (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft} \wedge \{\sigma(x)\}^{\triangleright} = \delta'(x)) \wedge \\
&\qquad\qquad b(\sigma(x_1),\ldots,\sigma(x_n)) = \mathtt{false}\} = \\
&\quad \{(\mathtt{B_1},\delta') \mid \exists \sigma \in \Sigma : \exists \delta \in B : \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta = \mathrm{dom}\ \delta' \wedge \\
&\qquad\qquad (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\}^{\triangleright} \sqsubseteq \delta(x) \wedge \{\sigma(x)\}^{\triangleright} = \delta'(x)) \wedge \\
&\qquad\qquad b(\sigma(x_1),\ldots,\sigma(x_n)) = \mathtt{true}\} \cup \\
&\quad \{(\mathtt{B_2},\delta') \mid \exists \sigma \in \Sigma : \exists \delta \in B : \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta = \mathrm{dom}\ \delta' \wedge \\
&\qquad\qquad (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\}^{\triangleright} \sqsubseteq \delta(x) \wedge \{\sigma(x)\}^{\triangleright} = \delta'(x)) \wedge \\
&\qquad\qquad b(\sigma(x_1),\ldots,\sigma(x_n)) = \mathtt{false}\} = \\
&\quad \{(\mathtt{B_1},\delta') \mid \exists \delta \in B : \delta' \sqsubseteq \delta \wedge \\
&\qquad\qquad (\exists (\alpha_1,\ldots,\alpha_n) \in D^n : b(\alpha_1,\ldots,\alpha_n) = \mathtt{true} \wedge \\
&\qquad\qquad\quad (\forall i \in \{1,\ldots n\} : \delta'(x_i) = \{\alpha_i\}^{\triangleright}))\} \cup \\
&\quad \{(\mathtt{B_2},\delta') \mid \exists \delta \in B : \delta' \sqsubseteq \delta \wedge \\
&\qquad\qquad (\exists (\alpha_1,\ldots,\alpha_n) \in D^n : b(\alpha_1,\ldots,\alpha_n) = \mathtt{false} \wedge \\
&\qquad\qquad\quad (\forall i \in \{1,\ldots n\} : \delta'(x_i) = \{\alpha_i\}^{\triangleright}))\} =
\end{aligned}
$$

Step 2. Again, we calculate the corresponding semantic rule for $L_2 = (Loc \nrightarrow \Sigma_L)$ with respect to L_1. For c as defined above and $\lambda \in \Sigma_L$ we set $B =_{\mathrm{def}} \{\delta \in \Sigma_L \mid \delta \sqsubseteq$

$\lambda\}$ and calculate according to the GC definitions on page 9:

$$\{c \mapsto \lambda\}^{\triangleleft} = \{(c, \delta) \mid \delta \in B\}$$

$$\{(c, \delta) \mid \delta \in B\} \longrightarrow_{L_1} q_1' \cup q_2'$$

$$q_1' =_{\text{def}} \{(\mathtt{B_1}, \delta') \mid \exists \delta \in B : \delta' \sqsubseteq \delta \wedge (\exists (\alpha_1, \ldots, \alpha_n) \in D^n : b(\alpha_1, \ldots, \alpha_n) = \mathtt{true} \wedge (\forall i \in \{1, \ldots n\} : \delta'(x_i) = \{\alpha_i\}^{\triangleright}))\}$$

$$q_2' =_{\text{def}} \{(\mathtt{B_2}, \delta') \mid \exists \delta \in B : \delta' \sqsubseteq \delta \wedge (\exists (\alpha_1, \ldots, \alpha_n) \in D^n : b(\alpha_1, \ldots, \alpha_n) = \mathtt{false} \wedge (\forall i \in \{1, \ldots n\} : \delta'(x_i) = \{\alpha_i\}^{\triangleright}))\}$$

From Theorem 1.7 we conclude

$$\{c \mapsto \lambda\} \longrightarrow_{L_2} (q_1' \cup q_2')^{\triangleright}$$

$$(q_1' \cup q_2')^{\triangleright} =$$

$$\{\mathtt{B_1} \mapsto \bigsqcup\{\delta' \mid \exists \delta \in B : \delta' \sqsubseteq \delta \wedge (\exists (\alpha_1, \ldots, \alpha_n) \in D^n : b(\alpha_1, \ldots, \alpha_n) = \mathtt{true} \wedge (\forall i \in \{1, \ldots n\} : \delta'(x_i) = \{\alpha_i\}^{\triangleright}))\},$$

$$\mathtt{B_2} \mapsto \bigsqcup\{\delta' \mid \exists \delta \in B : \delta' \sqsubseteq \delta \wedge (\exists (\alpha_1, \ldots, \alpha_n) \in D^n : b(\alpha_1, \ldots, \alpha_n) = \mathtt{false} \wedge (\forall i \in \{1, \ldots n\} : \delta'(x_i) = \{\alpha_i\}^{\triangleright}))\}\} =$$

$$\{\mathtt{B_1} \mapsto \bigsqcup\{\delta' \mid \delta' \sqsubseteq \lambda \wedge (\exists (\alpha_1, \ldots, \alpha_n) \in D^n : b(\alpha_1, \ldots, \alpha_n) = \mathtt{true} \wedge (\forall i \in \{1, \ldots n\} : \delta'(x_i) = \{\alpha_i\}^{\triangleright}))\},$$

$$\mathtt{B_2} \mapsto \bigsqcup\{\delta' \mid \delta' \sqsubseteq \lambda \wedge (\exists (\alpha_1, \ldots, \alpha_n) \in D^n : b(\alpha_1, \ldots, \alpha_n) = \mathtt{false} \wedge (\forall i \in \{1, \ldots n\} : \delta'(x_i) = \{\alpha_i\}^{\triangleright}))\}\} =$$

$$\{\mathtt{B_1} \mapsto \lambda_1, \mathtt{B_2} \mapsto \lambda_2\}$$

with $\text{dom}\ \lambda_1 = \text{dom}\ \lambda_2 = \text{dom}\ \lambda$ and

$$\forall x \in \text{dom}\ \lambda - \{x_1, \ldots, x_n\} : \lambda_1(x) = \lambda_2(x) = \lambda(x)$$

$$\forall i \in \{1, \ldots, n\} : \lambda_1(x_i) = \pi_i(((\lambda(x_1)^{\triangleleft} \times \ldots \times \lambda(x_n)^{\triangleleft}) \cap b^{-1}(\{\mathtt{true}\}))^{\triangleright})$$

$$\forall i \in \{1, \ldots, n\} : \lambda_2(x_i) = \pi_i(((\lambda(x_1)^{\triangleleft} \times \ldots \times \lambda(x_n)^{\triangleleft}) \cap b^{-1}(\{\mathtt{false}\}))^{\triangleright})$$

where $b^{-1}(\{\beta\}) =_{\text{def}} \{(\alpha_1, \ldots, \alpha_n) \mid b(\alpha_1, \ldots, \alpha_n) = \beta\}$. This proves (AIF3).

4. Validity of rule (AAS7). We proceed in an analogy to the proof for the rules in 1.

Step 1. We calculate the corresponding semantic rule for $L_1 = \mathbf{P}(Loc \times \Sigma_L)$ for a set $B \subseteq \Sigma_L$, using the definition of the GC between $\mathbf{P}(Loc \times \Sigma)$ and L_1:

$$\begin{aligned}
&\{((\mathtt{x_1[i]} = \mathtt{x_2; B}), \delta) \mid \delta \in B\}^{\triangleleft} = \\
&\quad \{((\mathtt{x_1[i]} = \mathtt{x_2; B}), \sigma) \mid \exists \delta \in B : \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta \wedge (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\} \\
&\quad \longrightarrow_{\mathbf{P}} p'
\end{aligned}$$

with

$$\begin{aligned}
p' =_{\mathrm{def}} \{(\mathtt{B}, \sigma[x_1 \mapsto \sigma(x_1)[\sigma(i) \mapsto \sigma(x_2)]]) \mid \exists \delta \in B : \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta \wedge \\
(\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\}
\end{aligned}$$

Since the transition relation $\longrightarrow_{L_1}$ has been introduced according to the rules of Theorem 1.7, we can apply rule (iii) of this theorem and conclude

$$\{((\mathtt{x_1[i]} = \mathtt{x_2; B}), \delta) \mid \delta \in B\} \longrightarrow_{L_1} p'^{\triangleright}$$

with $p'^{\triangleright}$ calculated according to the definition of $\_^{\triangleright}$ and by application of Lemma 1.8 as

$$\begin{aligned}
p'^{\triangleright} = \{(\mathtt{B}, \{y \mapsto \{\sigma(y)\}^{\triangleright} \mid y \in \mathrm{dom}\ \sigma - \{x_1\}\} \cup \{x_1 \mapsto \{\sigma(x_1)[\sigma(i) \mapsto \sigma(x_2)]\}^{\triangleright}\}) \mid \\
\exists \delta \in B : \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta \wedge (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\}
\end{aligned}$$

Step 2. We calculate the corresponding semantic rule for $L_2 = (Loc \nrightarrow \Sigma_L)$ as above in 1., Step 2, with $B =_{\mathrm{def}} \{\delta \in \Sigma_L \mid \delta \sqsubseteq \lambda\}$:

$$\begin{aligned}
&\{(\mathtt{x_1[i]} = \mathtt{x_2; B}) \mapsto \lambda\}^{\triangleleft} = \{((\mathtt{x_1[i]} = \mathtt{x_2; B}), \delta) \mid \delta \in B\} \\
&\{(\mathtt{x_1[i]} = \mathtt{x_2; B}, \delta) \mid \delta \in B\} \longrightarrow_{L_1} p'^{\triangleright} \\
&q' =_{\mathrm{def}} \{(\mathtt{B}, \{y \mapsto \{\sigma(y)\}^{\triangleright} \mid y \in \mathrm{dom}\ \sigma - \{x_1\}\} \cup \{x_1 \mapsto \{\sigma(x_1)[\sigma(i) \mapsto \sigma(x_2)]\}^{\triangleright}\}) \mid \\
&\qquad \exists \delta \in B : \mathrm{dom}\ \sigma = \mathrm{dom}\ \delta \wedge (\forall x \in \mathrm{dom}\ \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\}
\end{aligned}$$

From Theorem 1.10 we conclude

$$\{(\mathtt{x_1[i]} = \mathtt{x_2; B}) \mapsto \lambda\} \longrightarrow_{L_2} q'^{\triangleright}$$

and calculate according to the GC definitions on page 9

$$q'^{\triangleright} = \{\mathtt{B} \mapsto \lambda'\}$$

with dom $\lambda' = $ dom λ and $\forall y \in$ dom $\lambda' - \{x_1\} : \lambda'(y) = \lambda(y)$, as can be shown exactly as in 1. It remains to calculate the function value $\lambda'(x_1)$:

$$\lambda'(x_1) = \bigsqcup\{\{\sigma(x_1)[\sigma(i) \mapsto \sigma(x_2)]\}^{\triangleright} \mid$$
$$\exists \delta \in B : \text{dom } \sigma = \text{dom } \delta \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \delta(x)^{\triangleleft})\} =$$
$$\bigsqcup\{\{\sigma(x_1)[\sigma(i) \mapsto \sigma(x_2)]\}^{\triangleright} \mid \text{dom } \sigma \subseteq \text{dom } \lambda \wedge (\forall x \in \text{dom } \sigma : \{\sigma(x)\} \subseteq \lambda(x)^{\triangleleft})\} =$$
$$\bigsqcup\{\{\sigma(x_1)[\sigma(i) \mapsto \sigma(x_2)]\}^{\triangleright} \mid \sigma(x_1) \in \lambda(x_1)^{\triangleleft} \wedge \sigma(x_2) \in \lambda(x_2)^{\triangleleft} \wedge \sigma(i) \in \lambda(i)^{\triangleleft}\} =$$
$$f_{\mathbf{P}}{}^{\triangleright}$$

$$f_{\mathbf{P}} =_{\text{def}}$$
$$\{\sigma(x_1)[\sigma(i) \mapsto \sigma(x_2)] \mid \sigma(x_1) \in \lambda(x_1)^{\triangleleft} \wedge \sigma(x_2) \in \lambda(x_2)^{\triangleleft} \wedge \sigma(i) \in \lambda(i)^{\triangleleft}\}$$

From the GC definition for array variables given on page 10 we infer that dom $\lambda'(x_1) = $ dom $\lambda(x_1)$ and that for some $k_L \in$ dom $\lambda'(x_1)$

$$\lambda'(x_1)[k_L] =$$
$$f_{\mathbf{P}}{}^{\triangleright}[k_L] =$$
$$\bigsqcup\{\{\sigma(x_1)[\sigma(i) \mapsto \sigma(x_2)][k]\}^{\triangleright} \mid$$
$$\sigma(x_1) \in \lambda(x_1)^{\triangleleft} \wedge \sigma(x_2) \in \lambda(x_2)^{\triangleleft} \wedge \sigma(i) \in \lambda(i)^{\triangleleft} \wedge k \in k_L{}^{\triangleleft}\} =$$
$$\bigsqcup(\{\{\sigma(x_1)[k]\}^{\triangleright} \mid$$
$$\sigma(x_1) \in \lambda(x_1)^{\triangleleft} \wedge k \in k_L{}^{\triangleleft} \wedge (\exists \sigma(i) \in \lambda(i)^{\triangleleft} : k \neq \sigma(i))\} \cup$$
$$\{\{\sigma(x_2)\}^{\triangleright} \mid$$
$$\sigma(x_2) \in \lambda(x_2)^{\triangleleft} \wedge k \in k_L{}^{\triangleleft} \wedge (\exists \sigma(i) \in \lambda(i)^{\triangleleft} : k = \sigma(i))\}) =$$
$$\bigsqcup(\{\{\sigma(x_1)[k]\}^{\triangleright} \mid$$
$$\sigma(x_1) \in \lambda(x_1)^{\triangleleft} \wedge k \in k_L{}^{\triangleleft} \wedge (\#\lambda(i)^{\triangleleft} \geq 2 \vee k_L{}^{\triangleleft} \cap \lambda(i)^{\triangleleft} = \varnothing)\} \cup$$
$$\{\{\sigma(x_2)\}^{\triangleright} \mid \sigma(x_2) \in \lambda(x_2)^{\triangleleft} \wedge k_L{}^{\triangleleft} \cap \lambda(i)^{\triangleleft} \neq \varnothing\}) =$$
$$\bigsqcup(\{\{\sigma(x_1)[k]\}^{\triangleright} \mid$$
$$\sigma(x_1) \in \lambda(x_1)^{\triangleleft} \wedge k \in k_L{}^{\triangleleft} \wedge (\#\lambda(i)^{\triangleleft} \geq 2 \vee k_L \sqcap \lambda(i) = \bot)\} \cup$$
$$\{\{\sigma(x_2)\}^{\triangleright} \mid \sigma(x_2) \in \lambda(x_2)^{\triangleleft} \wedge k_L \sqcap \lambda(i) \neq \bot\}) =$$
$$\{\lambda(x_1)[k_L] \mid (\#\lambda(i)^{\triangleleft} \geq 2 \vee k_L \sqcap \lambda(i) = \bot)\} \sqcup \{\lambda(x_2) \mid k_L \sqcap \lambda(i) \neq \bot\}$$

Combining the results elaborated for $\lambda'(y)$ and $\lambda'(x_1)$ results in

$$\{(\mathtt{x_1[i]} = \mathtt{x_2;B}) \mapsto \lambda\} \longrightarrow_{L_2} \{\mathtt{B} \mapsto \lambda[x_1 \mapsto a']\}$$

with

$$\text{dom } a' = \text{dom } \lambda(x_1)$$
$$a'[k_L] = \{\lambda(x_1)[k_L] \mid (\#\lambda(i)^{\triangleleft} \geq 2 \vee k_L \sqcap \lambda(i) = \bot)\} \sqcup \{\lambda(x_2) \mid k_L \sqcap \lambda(i) \neq \bot\}$$

This proves (AAS7). □

References

[1] B. A. Davey and H. A. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, 2002.