CHAPTER 5

Computability in Higher Types

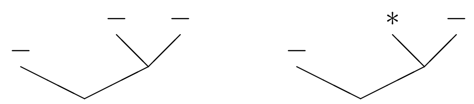
We view an infinite ("ideal") object as determined by the set of its finite approximations.

5.1. Binary trees

The algebra $\mathbb{Y}$ of binary trees is given by its two constructors, "$-$" for a leaf and "Branch" for a binary branch.

By a *token* we mean an (applicative) term with some argument positions occupied by $*$ (meaning "no knowledge"). An *extended token* is either a token of else $*$.

Examples:



The *height* of an extended token a^* is defined by $|-| := |*| := 0$ and $|\mathrm{Branch}(a_1^*, a_2^*)| := \max(|a_1^*|, |a_2^*|) + 1$.

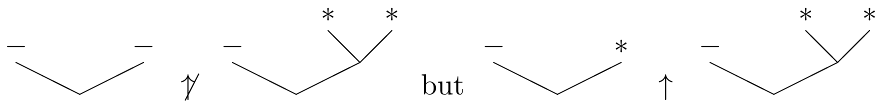
Definition. We define *consistency* $a \uparrow b$ of two tokens a, b inductively by the clauses

$$
\begin{aligned}
&- \uparrow - \\
(a_1 \uparrow b_1) \to (a_2 \uparrow b_2) &\to (\mathrm{Branch}(a_1, a_2) \uparrow \mathrm{Branch}(b_1, b_2)) \\
(a_1 \uparrow b_1) &\to (\mathrm{Branch}(a_1, *) \uparrow \mathrm{Branch}(b_1, b_2)) \\
(a_1 \uparrow b_1) &\to (\mathrm{Branch}(b_1, b_2) \uparrow \mathrm{Branch}(a_1, *)) \\
(a_2 \uparrow b_2) &\to (\mathrm{Branch}(*, a_2) \uparrow \mathrm{Branch}(b_1, b_2)) \\
(a_2 \uparrow b_2) &\to (\mathrm{Branch}(b_1, b_2) \uparrow \mathrm{Branch}(*, a_2)) \\
&\mathrm{Branch}(*, *) \uparrow \mathrm{Branch}(a, *) \\
&\mathrm{Branch}(a, *) \uparrow \mathrm{Branch}(*, *) \\
&\mathrm{Branch}(*, *) \uparrow \mathrm{Branch}(*, b) \\
&\mathrm{Branch}(*, b) \uparrow \mathrm{Branch}(*, *)
\end{aligned}
$$

This definition can easily be extended to consistency of finite sets U of tokens.

Examples for consistency:

Moreover



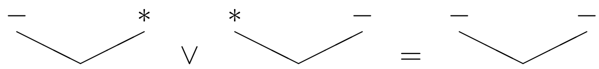
Definition. For any two consistent tokens a, b we inductively define a new token $a \vee b$ intended to be the least upper bound of the information contained in a and b. The clauses are

$$- \vee - := -$$

$$\mathrm{Branch}(a_1, a_2) \vee \mathrm{Branch}(b_1, b_2) := \mathrm{Branch}(a_1 \vee b_1, a_2 \vee b_2)$$

Here $a \vee *$ and $* \vee a$ both mean a, and $* \vee *$ means $*$.

Example:



Definition. For any two tokens a, b we inductively define a relation $a \geq b$ (*a entails b*) intended to mean that a contains at least the information contained in b. The clauses are

$$- \geq -$$

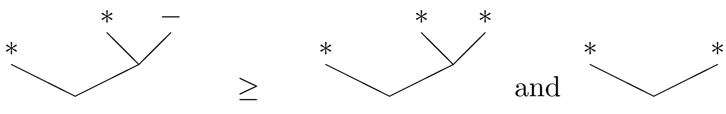
$$a_1 \geq b_1 \to a_2 \geq b_2 \to \mathrm{Branch}(a_1, a_2) \geq \mathrm{Branch}(b_1, b_2)$$

$$a_1 \geq b_1 \to \mathrm{Branch}(a_1, a_2) \geq \mathrm{Branch}(b_1, *)$$

$$a_2 \geq b_2 \to \mathrm{Branch}(a_1, a_2) \geq \mathrm{Branch}(*, b_2)$$

$$\mathrm{Branch}(a_1, a_2) \geq \mathrm{Branch}(*, *)$$

Examples of entailment



It is easy to see that $a \geq b$ implies $a \uparrow b$.

We now turn to possibly infinite (or "ideal") objects of our algebra $\mathbb{Y}$, which are viewed as given by their finite approximations, i.e., tokens.

Definition (Ideal in $\mathbb{Y}$). A possibly infinite set x of tokens in $\mathbb{Y}$ is called an *ideal* if

(a) x is consistent, i.e., any two tokens in x are consistent:

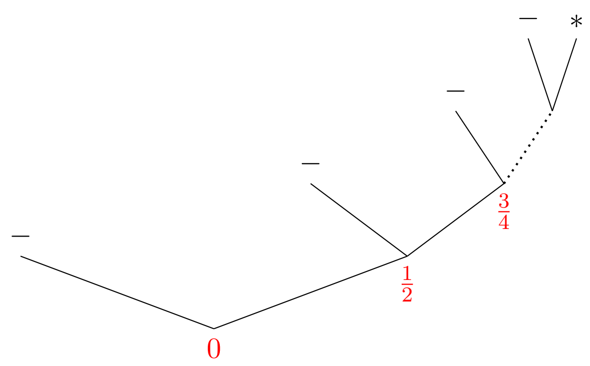
$$\forall_{a,b \in x}(a \uparrow b)$$

(b) x is deductively closed, i.e.,

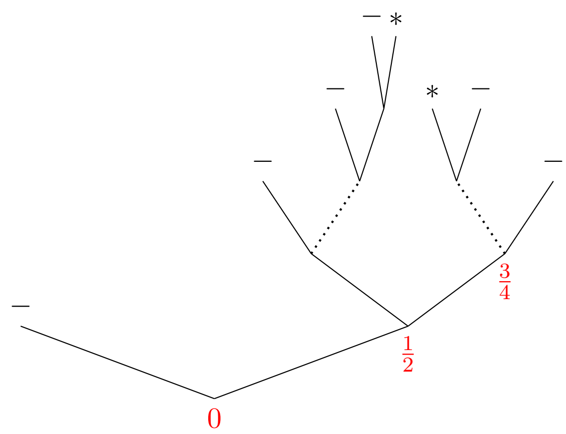
$$\forall_{a \in x} \forall_b (a \geq b \to b \in x)$$

Examples of ideals

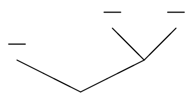
(1) $1 :=$ closure of all



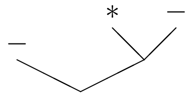
(2) -1 is defined similarly
(3) $-1 \cup 1$
(4) $\frac{1}{2} :=$ closure of all



(5) Closure of



(6) Closure of



Definition (Total and cototal ideals in $\mathbb{Y}$). An ideal x in $\mathbb{Y}$ is *total* if it contains a token built from constructors only, i.e., without $*$. We call x *cototal* if it is either total or else for every token $a \in x$ there is a token $b \in x$ with $b \geq a$ and b different from a.

Among the above examples of ideals

$(1) - (4)$	are cototal ideals,
(5)	is a total ideal,
(6)	is neither a total nor a cototal ideal.

5.2. Atomic coherent information systems

Definition. An *atomic coherent information system* (abbreviated *acis*) is a triple $(A, \mathrm{Con}, \geq)$ with A a countable set (the *tokens*, denoted $a, b, \dots$), Con a nonempty set of finite subsets of A (the *consistent* sets or *formal neighborhoods*, denoted $U, V, \dots$), and $\geq$ a transitive and reflexive relation on A (the *entailment relation*) which satisfy

(a) $\emptyset \in \mathrm{Con}$, and $\{a\} \in \mathrm{Con}$ for every $a \in A$,
(b) $U \in \mathrm{Con}$ if and only if every two-element subset of U is in Con, and
(c) if $\{a, b\} \in \mathrm{Con}$ and $b \geq c$, then $\{a, c\} \in \mathrm{Con}$.

We write $U \geq a$ for $\exists_{b \in U} b \geq a$, and $U \geq V$ for $\forall_{a \in V} U \geq a$.

Lemma. *Let $\boldsymbol{A} = (A, \mathrm{Con}, \geq)$ be an acis. $U \geq V_1, V_2$ implies $V_1 \cup V_2 \in \mathrm{Con}$.*

Proof. Let $b_1 \in V_1$, $b_2 \in V_2$. Then we have $a_1, a_2 \in U$ such that $a_i \geq b_i$. From $\{a_1, a_2\} \in \mathrm{Con}$ we obtain $\{a_1, b_2\} \in \mathrm{Con}$ by (c), hence $\{b_1, b_2\} \in \mathrm{Con}$ again by (c). $\square$

Definition. Let $\boldsymbol{A} = (A, \mathrm{Con}_A, \geq_A)$ and $\boldsymbol{B} = (B, \mathrm{Con}_B, \geq_B)$ be acis's. Define $\boldsymbol{A} \to \boldsymbol{B} = (C, \mathrm{Con}, \geq)$ by

$$C := \mathrm{Con}_A \times B,$$

$$\{(U_1, b_1), \dots (U_n, b_n)\} \in \mathrm{Con} :\leftrightarrow \forall_{i,j}\big(U_i \cup U_j \in \mathrm{Con}_A \to \{b_i, b_j\} \in \mathrm{Con}_B\big),$$

$$(U, b) \geq (V, c) :\leftrightarrow V \geq_A U \wedge b \geq_B c.$$

LEMMA. *Let* $\boldsymbol{A} = (A, \mathrm{Con}_A, \geq_A)$ *and* $\boldsymbol{B} = (B, \mathrm{Con}_B, \geq_B)$ *be acis's. Then* $\boldsymbol{A} \to \boldsymbol{B}$ *is an acis again.*

PROOF. Clearly $\geq$ is transitive and reflexive, and the conditions (a) and (b) of an acis hold; it remains to check (c). Let $\{(U_1, b_1), (U_2, b_2)\} \in$ Con and $(U_2, b_2) \geq (V, c)$, hence $V \geq U_2$ and $b_2 \geq c$. We must show $\{(U_1, b_1), (V, c)\} \in$ Con. Assume $U_1 \cup V \in$ Con; we must show $\{b_1, c\} \in$ Con. Now $U_1 \cup V \in$ Con and $V \geq U_2$ by the previous lemma imply $U_1 \cup U_2 \in$ Con. But then $\{b_1, b_2\} \in$ Con, hence $\{b_1, c\} \in$ Con by (c). □

Scott (1982) introduced the notion of an *approximable map* from $\boldsymbol{A}$ to $\boldsymbol{B}$. Such a map is given by a relation r between Con_A and B, where $r(U, b)$ intuitively means that whenever we are given the information $U \in \mathrm{Con}_A$ on the argument, then we know that at least the token b appears in the value.

DEFINITION (Approximable map). Let $\boldsymbol{A}$ and $\boldsymbol{B}$ be acis's. A relation $r \subseteq \mathrm{Con}_A \times B$ is an *approximable map* from $\boldsymbol{A}$ to $\boldsymbol{B}$ (written $r\colon \boldsymbol{A} \to \boldsymbol{B}$) if and only if

(a) if $r(U, b_1)$ and $r(U, b_2)$, then $\{b_1, b_2\} \in \mathrm{Con}_B$, and
(b) if $r(U, b)$, $V \geq_A U$ and $b \geq_B c$, then $r(V, c)$.

Call a (possibly infinite) set x of tokens *consistent* if $U \in$ Con for every finite subset $U \subseteq x$, and *deductively closed* if $\forall_{a \in x} \forall_{b \leq a} b \in x$. The *ideals* (or *objects*) of an information system are defined to be the consistent and deductively closed sets of tokens; we write $|\boldsymbol{A}|$ for the set of ideals of $\boldsymbol{A}$.

THEOREM 5.2.1. *Let* $\boldsymbol{A}$ *and* $\boldsymbol{B}$ *be acis's. The ideals of* $\boldsymbol{A} \to \boldsymbol{B}$ *are exactly the approximable maps from* $\boldsymbol{A}$ *to* $\boldsymbol{B}$.

PROOF. We show that $r \in |\boldsymbol{A} \to \boldsymbol{B}|$ satisfies the axioms for approximable maps. (a). Let $r(U, b_1)$ and $r(U, b_2)$. Then $\{b_1, b_2\} \in \mathrm{Con}_B$ by the consistency of r. (b). Let $r(U, b)$, $V \geq_A U$ and $b \geq_B c$. Then $(U, b) \geq (V, c)$ by definition, hence $r(V, c)$ by the deductive closure of r.

For the other direction suppose $r\colon \boldsymbol{A} \to \boldsymbol{B}$ is an approximable map. We must show that $r \in |\boldsymbol{A} \to \boldsymbol{B}|$. Consistency of r: Suppose $r(U_1, b_1)$, $r(U_2, b_2)$ and $U = U_1 \cup U_2 \in \mathrm{Con}_A$. We must show that $\{b_1, b_2\} \in \mathrm{Con}_B$. Now by definition of approximable maps, from $r(U_i, b_i)$ and $U \geq_A U_i$ we obtain $r(U, b_i)$, and hence $\{b_1, b_2\} \in \mathrm{Con}_B$. Deductive closure of r: Suppose $r(U, b)$ and $(U, b) \geq (V, c)$, i.e., $V \geq_A U \wedge b \geq_B c$. Then $r(V, c)$ by definition of approximable maps. □

The set $|\boldsymbol{A}|$ of ideals for $\boldsymbol{A}$ carries a natural topology (the Scott topology), which has the "cones" $\tilde{U} := \{\, z \mid z \supseteq U \,\}$ generated by the formal neighborhoods U as basis. The continuous maps $f\colon |\boldsymbol{A}| \to |\boldsymbol{B}|$ and the ideals

$r \in |\boldsymbol{A} \to \boldsymbol{B}|$ are in a bijective correspondence. With any $r \in |\boldsymbol{A} \to \boldsymbol{B}|$ we can associate a continuous $|r|\colon |\boldsymbol{A}| \to |\boldsymbol{B}|$:

$$|r|(z) := \{\, b \in B \mid r(U, b) \text{ for some } U \subseteq z \,\},$$

and with any continuous $f\colon |\boldsymbol{A}| \to |\boldsymbol{B}|$ we can associate $\hat{f} \in |\boldsymbol{A} \to \boldsymbol{B}|$:

$$\hat{f}(U, b) := (b \in f(\overline{U})).$$

These assignments are inverse to each other, i.e., $f = |\hat{f}|$ and $r = \widehat{|r|}$.

5.3. Partial continuous functionals

Recall that types are built from base types (free algebras, like $\mathbb{Y}$) by arrow formation $\tau \to \sigma$. For each type τ we can now construct an acis $\boldsymbol{C}_\tau$, in a natural way. For base types this has been done in Section 5.1 for the example of binary trees; it can be done similarly for arbitrary base types. For arrow types $\tau \to \sigma$ we define $\boldsymbol{C}_{\tau \to \sigma}$ to be $\boldsymbol{C}_\tau \to \boldsymbol{C}_\sigma$, as in Theorem 5.2.1. The ideals in $\boldsymbol{C}_\tau$ are called *partial continuous functionals* of type τ (Scott-Ershov model). Since ideals consist of tokens which can be coded by natural numbers, we have an easy way to define computability of these ideals:

Definition. A partial continuous functional $x \in |\boldsymbol{C}_\tau|$ is *computable* if it is recursively enumerable when viewed as a set of tokens.

We can now extend an arithmetical theory into a type theory with the Scott-Ershov model of partial continuous functional as the intended model, and moreover we have a reasonable notion of computability for such functionals.