1.5 Induction

More or less all sets of objects in computer science or logic are defined *inductively*. Typically, this is done in a bottom-up way, where starting with some definite set, it is closed under a given set of operations.

Example 1.5.1 (Inductive Sets)**.** In the following, some examples for inductively defined sets are presented:

1. The set of all Sudoku problem states, see Section 1.1, consists of the set of start states $(N; \top; \top)$ for consistent assignments N plus all states that can be derived from the start states by the rules Deduce, Conflict, Backtrack, and Fail. This is a finite set.

2. The set $\mathbb{N}$ of the natural numbers, consists of 0 plus all numbers that can be computed from 0 by adding 1. This is an infinite set.

3. The set of all strings Σ^* over a finite alphabet Σ. All letters of Σ are contained in Σ^* and if u and v are words out of Σ^* so is the word uv, see Section 1.2. This is an infinite set.

All the previous examples have in common that there is an underlying well-founded ordering on the sets induced by the construction. The minimal elements for the Sudoku are the problem states $(N; \top; \top)$, for the natural numbers it is 0 and for the set of strings it is the empty word. Now if we want to prove a property of an inductive set it is sufficient to prove it (i) for the minimal element(s) and (ii) assuming the property for an arbitrary set of elements, to prove that it holds for all elements that can be constructed "in one step" out those elements. This is the principle of *Noetherian Induction*.

Theorem 1.5.2 (Noetherian Induction)**.** Let $(M, \succ)$ be a well-founded ordering, and let Q be a predicate over elements of M. If for all $m \in M$ the implication

if $Q(m')$, for all $m' \in M$ so that $m \succ m'$,	(induction hypothesis)
then $Q(m)$.	(induction step)

is satisfied, then the property $Q(m)$ holds for all $m \in M$.

Proof. Let $X = \{m \in M \mid Q(m) \text{ does not hold}\}$. Suppose, $X \neq \emptyset$. Since $(M, \succ)$ is well-founded, X has a minimal element m_1. Hence for all $m' \in M$ with $m' \prec m_1$ the property $Q(m')$ holds. On the other hand, the implication which is presupposed for this theorem holds in particular also for m_1, hence $Q(m_1)$ must be true so that m_1 cannot be in X - a contradiction. □

Note that although the above implication sounds like a one step proof technique it is actually not. There are two cases. The first case concerns all elements that are minimal with respect to $\prec$ in M and for those the predicate Q needs to hold without any further assumption. The second case is then the induction step showing that by assuming Q for all elements strictly smaller than some m, we can prove it for m.

Now for context free grammars. *** Motivate Further *** Let $G = (N, T, P, S)$ be a context-free grammar (possibly infinite) and let q be a property of T^* (the words over the alphabet T of terminal symbols of G).

q holds for *all* words $w \in L(G)$, whenever one can prove the following two properties:

1. (*base cases*)
   $q(w')$ holds for each $w' \in T^*$ so that $X ::= w'$ is a rule in P.

2. (*step cases*)
   If $X ::= w_0 X_0 w_1 \dots w_n X_n w_{n+1}$ is in P with $X_i \in N$, $w_i \in T^*$, $n \geq 0$, then for all $w_i' \in L(G, X_i)$, whenever $q(w_i')$ holds for $0 \leq i \leq n$, then also $q(w_0 w_0' w_1 \dots w_n w_n' w_{n+1})$ holds.

Here $L(G, X_i) \subseteq T^*$ denotes the language generated by the grammar G from the nonterminal X_i.

Let $G = (N, T, P, S)$ be an *unambiguous* (why?) context-free grammar. A function f is well-defined on $L(G)$ (that is, unambiguously defined) whenever these 2 properties are satisfied:

1. (base cases)
   f is well-defined on the words $w' \in T^*$ for each rule $X ::= w'$ in P.

2. (step cases)
   If $X ::= w_0 X_0 w_1 \dots w_n X_n w_{n+1}$ is a rule in P then $f(w_0 w_0' w_1 \dots w_n w_n' w_{n+1})$ is well-defined, assuming that each of the $f(w_i')$ is well-defined.

Exercises

(1.19) Prove by Noetherian induction that for any $n \in \mathbb{N}^+$: $\Sigma_{i=1}^{n} i = \frac{n \cdot (n+1)}{2}$

1.6 Rewrite Systems

The final ingredient to actually start the journey through different logical systems is rewrite systems. Here I define the needed computer science background for defining algorithms in the form of rule sets. In Section 1.1 the rewrite rules Deduce, Conflict, Backtrack, and Fail defined an algorithm for solving 4×4 Sudokus. The rules operate on the set of Sudoku problem states, starting with a set of initial states $(N; \top; \top)$ and finishing either in a solution state $(N; D; \top)$ or a fail state $(N; \top; \bot)$. The latter are called *normal forms* (see below) with respect to the above rules, because no more rule is applicable to solution state $(N; D; \top)$ or a fail state $(N; \top; \bot)$.

Definition 1.6.1 (Rewrite System)**.** A *rewrite system* is a pair $(M, \to)$, where M is a set and $\to \subseteq M \times M$ is a binary relation on M. Figure 1.4 defines the needed notions for $\to$.

$\to^0$	$= \{ (a,a) \mid a \in M \}$	*identity*
$\to^{i+1}$	$= \to^i \circ \to$	$i+1$*-fold composition*
$\to^+$	$= \bigcup_{i>0} \to^i$	*transitive closure*
$\to^*$	$= \bigcup_{i\geq 0} \to^i = \to^+ \cup \to^0$	*reflexive transitive closure*
$\to^=$	$= \to \cup \to^0$	*reflexive closure*
$\to^{-1}$	$= \leftarrow = \{ (b,c) \mid c \to b \}$	*inverse*
$\leftrightarrow$	$= \to \cup \leftarrow$	*symmetric closure*
$\leftrightarrow^+$	$= (\leftrightarrow)^+$	*transitive symmetric closure*
$\leftrightarrow^*$	$= (\leftrightarrow)^*$	*refl. trans. symmetric closure*

Figure 1.4: Notation on $\to$

For a rewrite system $(M, \to)$ consider a sequence of elements a_i that are pairwise connected by the symmetric closure, i.e., $a_1 \leftrightarrow a_2 \leftrightarrow a_3 \ldots \leftrightarrow a_n$. We say that a_i is a *peak* in such a sequence, if actually $a_{i-1} \leftarrow a_i \to a_{i+1}$.

Actually, in Definition 1.6.1 I overload the symbol $\to$ that has already denoted logical implication, see Section 1.4, with a rewrite relation. This overloading will remain throughout this book. The rule symbol $\Rightarrow$ is only used on the meta level in this book, e.g., to define the Sudoku algorithm on problem states, Section 1.1. Nevertheless, this meta rule systems are also rewrite systems in the above sense. The rewrite symbol $\to$ is used on the formula level inside a problem state. This will become clear when I turn to more complex logics starting from Chapter 2.

Definition 1.6.2 (Reducible)**.** Let $(M, \to)$ be a rewrite system. An element $a \in M$ is *reducible*, if there is a $b \in M$ so that $a \to b$. An element $a \in M$ is *in normal form (irreducible)*, if it is not reducible. An element $c \in M$ is a *normal form* of b, if $b \to^* c$ and c is in normal form, notated $c = b{\downarrow}$ (if the normal form of b is unique). Two elements b and c are *joinable*, if there is an a so that $b \to^* a \ {}^*\!\leftarrow c$, notated $b \downarrow c$.

Definition 1.6.3 (Properties of $\to$)**.** A relation $\to$ is called

Church-Rosser	if $b \leftrightarrow^* c$ implies $b \downarrow c$
confluent	if $b \ {}^*\!\leftarrow a \to^* c$ implies $b \downarrow c$
locally confluent	if $b \leftarrow a \to c$ implies $b \downarrow c$
terminating	if there is no infinite descending chain $b_0 \to b_1 \ldots$
normalizing	if every $b \in A$ has a normal form
convergent	if it is confluent and terminating

Lemma 1.6.4. If $\to$ is terminating, then it is normalizing.

The reverse implication of Lemma 1.6.4 does not hold. Assuming this is a frequent mistake. Consider $M = \{a, b, c\}$ and the relation $a \to b$, $b \to a$, and $b \to c$. Then $(M, \to)$ is obviously not terminating, because we can cycle between a and b. However, $(M, \to)$ is normalizing. The normal form is c for all elements of M. Similarly, there are rewrite systems that are locally confluent, but not confluent, see Figure **??**. *** to be done *** In the context of termination the property holds, see Lemma 1.6.6.

Theorem 1.6.5. The following properties are equivalent for any rewrite system $(S, \to)$:

(i) $\to$ has the Church-Rosser property.

(ii) $\to$ is confluent.

Proof. (i) $\Rightarrow$ (ii): trivial.

(ii) $\Rightarrow$ (i): by induction on the number of peaks in the derivation $b \leftrightarrow^* c$. □

Lemma 1.6.6 (Newman's Lemma [**?**]: Confluence versus Local Confluence)**.** Let $(S, \to)$ be a terminating rewrite system. Then the following properties are equivalent:

(i) $\to$ is confluent

(ii) $\to$ is locally confluent

Proof. (i) $\Rightarrow$ (ii): trivial.

(ii) $\Rightarrow$ (i): Since $\to$ is terminating, it is a well-founded ordering (see Exercise **??**). This justifies a proof by Noetherian induction where the property $Q(a)$ is "a is confluent". Applying Noetherian induction, confluence holds for all $a' \in M$ with $m \to^+ a'$ and needs to be shown for a. Consider the confluence property for a: $b \,{}^*\!\leftarrow m \to^* c$. If $b = a$ or $c = a$ the proof is done. For otherwise, the situation is in more detail $b \,{}^*\!\leftarrow b' \leftarrow a \to c' \to^* c$. By local confluence there is an a' with $b' \to^* a' \,{}^*\!\leftarrow c'$. Now a', b, c are strictly smaller than a, they are confluent and hence can be rewritten so a single a'', finishing the proof. □

Lemma 1.6.7. If $\to$ is confluent, then every element has at most one normal form.

Proof. Suppose that some element $a \in A$ has normal forms b and c, then $b \,{}^*\!\leftarrow a \to^* c$. If $\to$ is confluent, then $b \to^* d \,{}^*\!\leftarrow c$ for some $d \in A$. Since b and c are normal forms, both derivations must be empty, hence $b \to^0 d \,{}^0\!\leftarrow c$, so b, c, and d must be identical. □

Corollary 1.6.8. If $\to$ is normalizing and confluent, then every element b has a unique normal form.

Proposition 1.6.9. If $\to$ is normalizing and confluent, then $b \leftrightarrow^* c$ if and only if $b{\downarrow} = c{\downarrow}$.

Proof. Either using Theorem 1.6.5 or directly by induction on the length of the derivation of $b \leftrightarrow^* c$. □

Exercises

(1.20) Prove Corollary 1.6.8.

(1.21) Prove Proposition 1.6.9 by induction on the length of the derivation without using the Church-Rosser Theorem.

(1.22)* A relation $\rightarrow$ is *semi-confluent* iff

$$y_1 \leftarrow x \rightarrow^* y_2 \Rightarrow y_1 \downarrow y_2$$

Prove: A relation $\rightarrow$ is semi-confluent iff it is confluent.

(1.23)* A relation $\rightarrow$ is *strongly confluent* (for all x, y_1, y_2) iff

$$y_1 \leftarrow x \rightarrow y_2 \Rightarrow \exists z.\ y_1 \rightarrow^* z \leftarrow^= y_2$$

Does the strong confluence imply the following property?

$$y_1 \leftarrow x \rightarrow y_2 \Rightarrow \exists z.\ y_1 \rightarrow^= z \leftarrow^= y_2$$

Give a proof or counterexample.

(1.24) Prove that the following term rewrite system is confluent:

$$\begin{array}{rcl} f(g(x)) & \rightarrow & x \\ g(f(x)) & \rightarrow & x \\ f(b) & \rightarrow & c \\ b & \rightarrow & g(c) \end{array}$$

(1.25) Is the rewrite system

$$\{\, f(a) \rightarrow f(b),\ f(b) \rightarrow f(c),\ f(c) \rightarrow f(a),\ f(x) \rightarrow x \,\}$$

(i) terminating, (ii) normalizing, (iii) locally confluent, (iv) confluent? Give a brief explanation.

(1.26) Prove or refute the following statement. There exists a rewrite system $(M, \rightarrow)$ so that every $a \in M$ has exactly two normal forms.

(1.27) Given the rewrite system

$$\begin{array}{rll} R: & x + 0 \rightarrow x & 0 + x \rightarrow x \\ & x + (-x) \rightarrow 0 & (-x) + x \rightarrow 0 \\ & -0 \rightarrow 0 & -(-x) \rightarrow x \\ & -(x + y) \rightarrow (-x) + (-y) & (x + y) + z \rightarrow x + (y + z) \\ & x + ((-x) + y) \rightarrow y & (-x) + (x + y) \rightarrow y \end{array}$$

compute the rewrite successors of $s = -((-x) + (y + x))$ and $t = ((-x) + (-y)) + x$.

(1.28)* Show that the following property holds: Let $\rightarrow_1$ and $\rightarrow_2$ be two binary relations over M, so that $(\rightarrow_1 \cup \rightarrow_2)$ is transitive. Then $(\rightarrow_1 \cup \rightarrow_2)$ is terminating if and only if $\rightarrow_1$ and $\rightarrow_2$ are terminating. (Hint: Start with the assumption that there is an infinite $(\rightarrow_1 \cup \rightarrow_2)$ chain.)