First Lectures of Automata Theory and Formal Languages

1. Preliminaries

Unbounded counting:

$x = x + 1$

$x = (((\ (((\ (((x + 1)))\ )))\ )))$

Compiler for recursion:

$f(x) = \textbf{if } x = 0 \textbf{ then } 1 \textbf{ else } x \times f(x - 1)$

$f(5) = 120$

Compiler for recursion again (types are important):

$f(x, g) = \textbf{if } x = 0 \textbf{ then } 1 \textbf{ else } x \times g(x - 1)$

$f(5, f) = 120$

Call-by-name and call-by-value:

$f(x) = 1 \quad g(x) = g(x) + 3$

$f(g(2)) = \ldots$ *call-by-name*: 1 *call-by-value*: nontermination

Java, Python, C++ are call-by-value languages.

Haskell is call-by-name.

2. Three stars: S^*, $\rightarrow^*$, b^*

symbols, characters: $\Sigma = \{a, b\}$

words, strings, sequences: ab, $abba$, ε, bba, $\ldots$

concatenation of words: $ab \cdot abba = ababba$

set S of words

concatenation of sets of words: $S_1 \cdot S_2 = \{w_1 \cdot w_2 \mid w_1 \in S_1 \text{ and } w_2 \in S_2\}$

$S^0 = \{\epsilon\}$

$S^1 = S$

$S^{i+1} = S^i \cdot S = S \cdot S^i$

$S^* = \{\varepsilon\} \cup S \cup S \cdot S \cup \ldots = \bigcup_{i \geq 0} S^i$ (first star: S^*)

$S^+ = S \cup S \cdot S \cup \ldots = \bigcup_{i \geq 1} S^i$

THEOREM 1. For any set S of words, if $\varepsilon \in S$ then $S^* = S \cdot S^* = S^+$.

Grammar: $G = \langle V_T, V_N, P, S \rangle$

terminal alphabet: V_T

nonterminal alphabet: V_N

alphabet: $V = V_T \cup V_N$

set P of productions: set of pairs $\alpha \rightarrow \beta$, where $\alpha \in V^+$ and $\beta \in V^*$

axiom: $S \in V_N$

Type 2 grammars: $\alpha \in V_N$

Example: $S \rightarrow SA \mid a \qquad A \rightarrow aAa \mid bA \mid \varepsilon$

Language of a grammar: $L(G) = L(S) = \{w \mid S \to^* w \text{ and } w \in V_T^*\}$
$w_1 \to w_2$ iff $\exists u, v, \alpha, \beta$ such that $w_1 = u\,\alpha\,v$ and $w_2 = u\,\beta\,v$ and $\alpha \to \beta \in P$.

$\to^*$ is the reflexive, transitive closure of $\to$. (second star: $\to^*$)

Language of a nonterminal A: $L(A) = \{w \mid A \to^* w \text{ and } w \in V_T^*\}$

$s \in V^*$ is a *sentential form derived from* A if $A \to^* s$.

Take any $b \in V_T$. $b^* = \{\varepsilon, b, bb, \ldots\} = \{b^0, b^1, b^2, \ldots\} = \bigcup_{i \geq 0}\{b^i\}$. (third star: b^*)

Notation. $\{a, b\}^* = (a+b)^*$: all words of any length with characters a or b only.

Arithmetic Expressions with precedence of × *over* +.

$$\begin{array}{lcllllllll}
E & \to & E+T & \mid & E-T & \mid & T & & & \text{(expressions)}\\
T & \to & T \times F & \mid & T/F & \mid & F & & & \text{(terms)}\\
F & \to & a & \mid & b & \mid & c & \mid & (E) & \text{(factors)}
\end{array}$$

terminal alphabet: $V_T = \{a, b, c, +, -, \times, /, (,)\}$

non terminal alphabet: $V_N = \{E, T, F\}$

axiom: E.

The expression $a+c\times b$ has the *parse tree* (which shows the precedence of the operators + and ×) (downlines denote the relation $\to$) depicted in Figure 1.

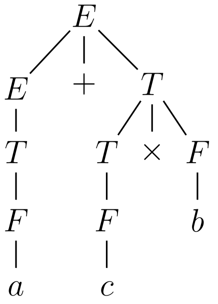

FIGURE 1. Parse tree of the arithmetic expression $a+c\times b$. Precedence of × over + is realized by the bottom-up evaluation of the tree: before computing E as $E+T$, we have to compute T as $T \times F$.

Arithmetic Expressions without precedence of × *over* +.

$$E \;\to\; E+E \;\mid\; E-E \;\mid\; E\times E \;\mid\; E/E \;\mid\; a \;\mid\; b \;\mid\; c \;\mid\; (E)$$

The expression $a+c\times b$ has the two parse trees (and thus this grammar is ambiguous) depicted in Figure 2.

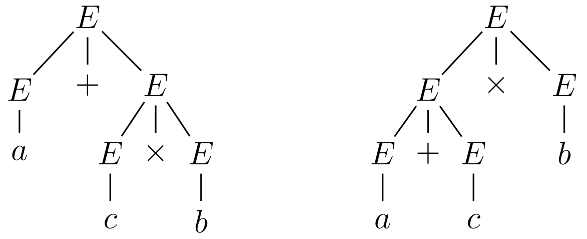

FIGURE 2. The two parse trees of the arithmetic expression $a + c \times b$, which correspond to $a + (c \times b)$ and $(a + c) \times b$, respectively. Precedence of × over + should be enforced by using parentheses.

3. Grammars for Programming Languages

Identifiers.

$$\begin{array}{lcll} Id & \rightarrow & L\,A & \text{(identifier)} \\ A & \rightarrow & L\,A \;\mid\; D\,A \;\mid\; \varepsilon & \\ L & \rightarrow & a \,|\, A \,|\, b \,|\, B \,|\, \ldots & \text{(letter)} \\ D & \rightarrow & 0 \,|\, 1 \,|\, 2 \,|\, 3 \,|\, \ldots \,|\, 9 & \text{(digit)} \end{array}$$

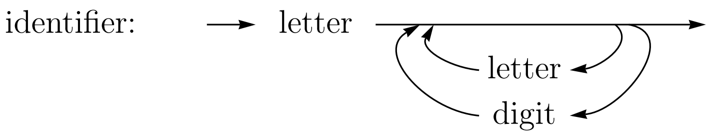

FIGURE 3. Productions represented by diagrams. Identifiers (*Id*), letter (*L*), and digit (*D*).

HTML.

Appearance of the text:

The things I *hate*:
1. Moldy bread.
2. People who drive too slow in the fast lane.

HTML source:
(P = Paragraph, EM = Emphasize, OL = Ordered List, LI = List Item)

```
<P> The things I <EM> hate </EM>:
<OL>
<LI> Moldy bread.
<LI> People who drive too slow in the fast lane.
</OL>
```

Grammar:

$V_T = \{a, A, b, B, \ldots,$ <EM>, </EM>, <OL>, </OL>, <P>, <LI>$\}$

$V_N = \{\mathit{Char}, \mathit{Text}, \mathit{Doc}, \mathit{Elem}, \mathit{List}, \mathit{Item}\}$

The axiom of the grammar is *Doc*.

The productions are:

$$\begin{array}{lll}
\mathit{Doc} & \rightarrow & \varepsilon \mid \mathit{Elem}\ \mathit{Doc} \\
\mathit{Elem} & \rightarrow & \mathit{Text} \mid \texttt{<}\mathrm{EM}\texttt{>}\ \mathit{Doc}\ \texttt{<}/\mathrm{EM}\texttt{>} \mid \texttt{<}\mathrm{P}\texttt{>}\ \mathit{Doc}\ \mid \texttt{<}\mathrm{OL}\texttt{>}\ \mathit{List}\ \texttt{<}/\mathrm{OL}\texttt{>} \\
\mathit{List} & \rightarrow & \varepsilon \mid \mathit{Item}\ \mathit{List} \\
\mathit{Item} & \rightarrow & \texttt{<}\mathrm{LI}\texttt{>}\ \mathit{Doc} \\
\mathit{Char} & \rightarrow & a \mid A \mid B \mid b \mid \ldots \\
\mathit{Text} & \rightarrow & \varepsilon \mid \mathit{Char}\ \mathit{Text}
\end{array}$$

4. Mathematical Induction

Natural Numbers.

$N = \{0, s(0), s(s(0)), \ldots\}$.
Consider a predicate $P(n)$ over the set N, that is, a subset of N.

(Mathematical Induction) (†)

$$\frac{P(0) \qquad \forall n \in N.\, (P(n) \Rightarrow P(s(n)))}{\forall k \in N.\, P(k)}$$

In order to avoid *regressio ad infinitum*, the premise $\forall n \in N.\, (P(n) \Rightarrow P(s(n)))$, which stands for $\forall n.\, \big(n \in N \Rightarrow (P(n) \Rightarrow P(s(n)))\big)$, can be proved by using the Generalization Rule of First Order Predicate Calculus:

$$\text{(Generalization Rule)} \qquad \frac{A(x)}{\forall x.\, A(x)}$$

Indeed, by the Generalization Rule from $n \in N \Rightarrow (P(n) \Rightarrow P(s(n)))$ we get:
$\forall n.\, \big(n \in N \Rightarrow (P(n) \Rightarrow P(s(n)))\big)$.

For a predicate $P(m,n)$ with two arguments over the set $N \times N$ we have the following rule:

(Mathematical Induction with two arguments) (††)

$$\frac{P(0,0) \quad \forall n \in N.\, (P(0,n) \Rightarrow P(0,s(n))) \qquad \forall m,n \in N.\, (P(m,n) \Rightarrow P(s(m),n))}{\forall h,k \in N.\, P(h,k)}$$

We can derive this rule from the mathematical induction rule for predicates with one argument (see rule (†) above) as follows.

Consider the predicate $Q(m) =_{def} \forall n \in N.\, P(m,n)$. In order to show that $\forall m \in N.\, Q(m)$ by mathematical induction we need to show:

(i) $Q(0)$ and (ii) $\forall m \in N.\, Q(m) \Rightarrow Q(s(m))$.

(i) In order to show $Q(0)$, that is, $\forall n \in N.\, P(0,n)$, by mathematical induction for predicates with one argument only, we need to show:

(α) $P(0,0)$ and

(β) $\forall n \in N.\, (P(0,n) \Rightarrow P(0,s(n)))$.

(ii) In order to show $\forall m\!\in\! N.\, Q(m) \Rightarrow Q(s(m))$, that is,

$(\gamma')\ \forall m\!\in\! N.\, \big(\forall n\!\in\! N.\, P(m,n)\big) \Rightarrow \big(\forall n\!\in\! N.\, P(s(m),n)\big)$,

it is enough to show:

$(\gamma)\ \forall m,n\!\in\! N.\, (P(m,n) \Rightarrow P(s(m),n))$,

because for all binary predicates A and B, we have that $\forall m,n.\,(A(m,n) \Rightarrow B(m,n))$ implies $\forall m.\,(\forall n.\, A(m,n)) \Rightarrow (\forall n.\, B(m,n))$.

Remark 2. Note that "something funny happens to the left of the arrow" (as Prof. John Reynolds says), that is, if $A \Rightarrow B$, then $(B \Rightarrow C) \Rightarrow (A \Rightarrow C)$. Since $\forall m,n.\,(A(m,n) \Rightarrow B(m,n))$ implies $\forall m.\,(\forall n.\, A(m,n)) \Rightarrow (\forall n.\, B(m,n))$, we have that the Mathematical Induction rule (†) implies the Mathematical Induction rule with two arguments (††). □

Exercise 3. Prove by mathematical induction that, for all $n\geq 0$,

(i) $\sum_{i=0}^{n} i = \dfrac{n\ (n+1)}{2}$.

Theorem 4. [**Commutativity of Plus**] For all natural numbers m and n, we have that $m+n = n+m$, where the + operation is defined by primitive recursion as follows:

$$\begin{array}{lll} (P0) & \forall n. & 0+n \;= n \\ (Ps) & \forall m,n. & s(m)+n = s(m+n) \end{array}$$

Proof. By mathematical induction for a predicate with two arguments. We have to prove the following three facts:

$$\begin{array}{llll} (F1) & & 0+0 = 0+0 & \\ (F2) & \forall n. & 0+n = n+0 & \Rightarrow\ 0+s(n) = s(n)+0 \\ (F3) & \forall m,n. & m+n = n+m & \Rightarrow\ s(m)+n = n+s(m) \end{array}$$

Fact $(F1)$ follows from $(P0)$ because both sides are equal to 0. For Fact $(F2)$ let us consider a generic value n and the two sides $0+s(n)$ and $s(n)+0$ of the conclusion.

Now, $0+s(n) = \{\text{by } (P0)\} = s(n)$, and

$s(n)+0 = \{\text{by } (Ps)\} =$

$= s(n+0) = \{\text{by inductive hypothesis (see premise } 0+n=n+0 \text{ in Fact } (F2))\} =$

$= s(0+n) = \{\text{by } (P0)\} = s(n)$.

Since n is a generic value, by generalization we get Fact $(F2)$.

For Fact $(F3)$ let us consider the two generic values m and n. We have that:

$s(m)+n = \{\text{by } (Ps)\} =$

$= s(m+n) = \{\text{by inductive hypothesis (see premise } m+n=n+m \text{ in Fact } (F3))\} =$

$= s(n+m) = \{\text{by Lemma 5 below}\} = n+s(m)$.

Since m and n are generic values, by generalization we get Fact $(F3)$.

Lemma 5. For all natural numbers m and n, we have that $s(m+n) = m+s(n)$.

Proof. We proceed by mathematical induction on m.

(*Basis*) $m=0$. We have to show that $\forall n\!\in\! N.\, s(0+n) = 0+s(n)$. Indeed, by $(P0)$ the left hand side (l.h.s.) $s(0+n)$ and the right hand side (r.h.s.) $0+s(n)$ are both equal to $s(n)$.

(*Step*) We have to show that

$\forall m \in N. (\forall n \in N. s(m+n) = m+s(n)) \;\Rightarrow\; (\forall n \in N,\; s(s(m)+n) = s(m)+s(n))$.

Let us consider a generic value m. We assume that $\forall n \in N. s(m+n) = m+s(n)$ and we have to show that $\forall n \in N. s(s(m)+n) = s(m)+s(n)$.

Let us consider a generic value n. For l.h.s.: $s(s(m)+n) = \{\text{by } (Ps)\} = s(s(m+n))$. For r.h.s.: $s(m)+s(n) = \{\text{by } (Ps)\} = s(m+s(n)) = \{\text{by inductive hypothesis}\} = s(s(m+n))$. □

Complete Induction for natural numbers.

(Complete Induction)

$$\frac{\forall n \in N. ((\forall h \in N. h < n \Rightarrow P(h)) \Rightarrow P(n))}{\forall k \in N. P(k)}$$

THEOREM 6. Every non-prime number is a product of prime numbers.

PROOF. By complete induction. Consider a non-prime number m. It has at least a factor f different from 1 and m. The two factors f and $\frac{m}{f}$, being both less than m, by complete induction, are products of prime numbers.

Assume that $f = p_1^{i_1}\, p_2^{i_2} \ldots p_k^{i_k}$ and $\frac{m}{f} = q_1^{j_1}\, q_2^{j_2} \ldots q_h^{j_h}$, where the p_i's and the q_j's are all prime numbers. Thus, $m = p_1^{i_1}\, p_2^{i_2} \ldots p_k^{i_k}\, q_1^{j_1}\, q_2^{j_2} \ldots q_h^{j_h}$. (Note that in this product, if we replace every subproduct of the form $p_r^{i_r}\, q_s^{j_s}$, with $p_r = q_s$, by $p_r^{i_r+j_s}$, we get a factorization of m such that $m = p_1^{u_1}\, p_2^{u_2} \ldots p_t^{u_t}$, where the prime numbers $p_1, p_2, \ldots, p_t$ are all distinct).

□

5. Equivalence of Complete Induction and the Least Number Principle

In this section we will prove that from Complete Induction one can derive the Least Number Principle. The Least Number Principle states that:

"if a property holds for one or more natural numbers,
then there exists a minimum natural number for which it holds",

that is, using a formula of the first order predicate calculus,

$\exists y.\, \psi(y) \;\Rightarrow\; \big(\exists z. ((\forall x < z. \neg\psi(x)) \wedge \psi(z))\big)$.

In this formula and in all formulas of our proof, we assume that the quantified variables all range over the natural numbers. Thus, for instance,

$\forall x. \varphi(x)$ stands for $\forall x. x \in N \Rightarrow \varphi(x)$.

In our proof we have indicated between curly brackets the reason which justifies the step of the proof. The Complete Induction Principle can be stated as follows:

$\big(\forall z. ((\forall x. (x < z) \Rightarrow \varphi(x)) \Rightarrow \varphi(z))\big) \;\Rightarrow\; \forall y.\, \varphi(y)$

which can be abbreviated as

$\big(\forall z. ((\forall x < z. \varphi(x)) \Rightarrow \varphi(z))\big) \;\Rightarrow\; \forall y.\, \varphi(y)$

$\{\text{by } (a \Rightarrow b) \Leftrightarrow (\neg a \vee b)\}$

$$\big(\forall z.(\neg(\forall x\!<\!z.\,\varphi(x)) \vee \varphi(z))\big) \;\Rightarrow\; \forall y.\,\varphi(y)$$
$$\{\text{by } \varphi(x) \Leftrightarrow \neg\psi(x)\}$$
$$\big(\forall z.(\neg(\forall x\!<\!z.\,\neg\psi(x)) \vee \neg\psi(z))\big) \;\Rightarrow\; \forall y.\,\neg\psi(y)$$
$$\{\text{by } (a \Rightarrow b) \;\Leftrightarrow\; (\neg b \Rightarrow \neg a)\}$$
$$\neg\forall y.\,\neg\psi(y) \;\Rightarrow\; \neg\big(\forall z.(\neg(\forall x\!<\!z.\,\neg\psi(x)) \vee \neg\psi(z))\big)$$
$$\{\text{by } \neg\forall x.\psi(x) \Leftrightarrow \exists x.\neg\psi(x)\}$$
$$\exists y.\,\neg\neg\psi(y) \;\Rightarrow\; \big(\exists z.\neg(\neg(\forall x\!<\!z.\,\neg\psi(x)) \vee \neg\psi(z))\big)$$
$$\{\text{by } \neg(a \vee b) \;\Leftrightarrow\; (\neg a \wedge \neg b)\text{, that is, by the De Morgan law})$$
$$\exists y.\,\neg\neg\psi(y) \;\Rightarrow\; \big(\exists z.(\neg\neg(\forall x\!<\!z.\,\neg\psi(x)) \wedge \neg\neg\psi(z))\big)$$
$$\{\text{by } \neg\neg a \;\Leftrightarrow\; a\text{, that is, by the `tertium non datur' law})$$
$$\exists y.\,\psi(y) \;\Rightarrow\; \big(\exists z.((\forall x\!<\!z.\,\neg\psi(x)) \wedge \psi(z))\big)$$

which states the Least Number Principle.

6. Five Proofs

Consider the grammar with axiom S and productions:

$S \rightarrow SA \mid a \qquad A \rightarrow a\,A\,a \mid b\,A \mid \varepsilon$

Lemma 7. $b^* \subseteq L(A)$.

Proof. We show that for all $n \geq 0$, $b^n \in L(A)$ by induction on n.
(*Basis*) $n = 0$. To show: $\varepsilon \in L(A)$. Indeed: $A \rightarrow \varepsilon$.
(*Step*) Assume $b^n \in L(A)$. To show: $b^{n+1} \in L(A)$.

Indeed: $b^{n+1} = b\,b^n$ and $b\,b^n \in b\,L(A)$ because, by inductive hypothesis, $b^n \in L(A)$. Then, by the production $A \rightarrow bA$, we get: $b\,L(A) \subseteq L(A)$, and thus $b\,b^n \in L(A)$. □

Lemma 8. $a\,b^*\,a \subseteq L(A)$.

Proof. We show that for all $n \geq 0$, $a\,b^n\,a \in L(A)$ by induction on n.
(*Basis*) $n = 0$. To show: $a\,a \in L(A)$.

Indeed: $a\,a \in a\,L(A)\,a$ $\{$because $A \rightarrow \varepsilon\}$ $\subseteq L(A)$ $\{$because $A \rightarrow a\,A\,a\}$.
(*Step*) Assume $a\,b^n\,a \in L(A)$. To show: $a\,b^{n+1}\,a \in L(A)$.

Indeed: $a\,b^{n+1}\,a = a\,b\,b^n\,a$ and, by Lemma 7, $a\,b\,b^n\,a \in a\,L(A)\,a$. Then, by the production $A \rightarrow aAa$, we get: $a\,L(A)\,a \subseteq L(A)$, and thus $a\,b\,b^n\,a \in L(A)$. □

Lemma 9. For any *sentential form* s such that $A \rightarrow^* s$, s has an even number of a's.

Proof. By induction on the number n of applications of $\rightarrow$.
(*Basis*) $n = 0$. To show: A has an even number of a's. Obvious, because A has 0 a's.
(*Step*) Assume s such that: (i) $A \rightarrow^n s$, and (ii) s has an even number of a's. To show: for any s such that $A \rightarrow^{n+1} s$, s has an even number of a's.
Take any s such that $A \rightarrow^{n+1} s$ and s has an even number of a's. We have that there exists a sentential form s_1 such that $A \rightarrow^n s_1 \rightarrow s$. By induction hypothesis s_1 has an even number of a's. The rightmost rewriting $\rightarrow$ can be either: (i) $A \rightarrow a\,A\,a$, or (ii) $A \rightarrow b\,A$, or (iii) $A \rightarrow \varepsilon$. In Case (i) the number of a's in s is even because if k is

even, then $k+2$ is even. In Cases (ii) and (iii) the number of a's is not changed and, thus, it is even by inductive hypothesis. □

Let $(a+b)^*_{ea}$ denotes the set of words in $(a+b)^*$ that have an even number of a's.

THEOREM 10. $(L(A))^* = (a+b)^*_{ea}$.

PROOF. We have to show: Point (1): $(L(A))^* \subseteq (a+b)^*_{ea}$, and
Point (2): $(a+b)^*_{ea} \subseteq (L(A))^*$.

Point (1) is a consequence of the following points: (i) every $w \in L(A)$ is a particular sentential form derived from A, (ii) Lemma 9, (iii) for every $n \geq 0$, every word in $(L(A))^n$ is either ε or the concatenation of words in $L(A)$, and (iv) for all $n \geq 0$, the sum of n even numbers is an even number.

Point (2) is proved by induction on k, defined as the half of the number of a's in a word of $(a+b)^*_{ea}$.

(*Basis*) $k=0$. To show: $b^* \subseteq (L(A))^*$.

Indeed, by Lemma 7, $b^* \subseteq L(A) \subseteq (L(A))^*$.

(*Step*) Assume $\forall w. (w \in (a+b)^*$ and w has $2k$ a's$) \Rightarrow w \in (L(A))^*$ (inductive hypothesis).

To show $\forall w. (w \in (a+b)^*$ and w has $2(k{+}1)$ a's$) \Rightarrow w \in (L(A))^*$.

Indeed, take any word $w \in (a+b)^*$ such that w has $2(k{+}1)$ a's. w is of the form: $v\, b^k a\, b^m a\, b^n$, for some $k, m, n \geq 0$ and some $v \in (a+b)^*$ such that v has $2k$ a's.

By inductive hypothesis $v \in (L(A))^*$. By Lemma 7, $b^k \in L(A)$. By Lemma 8, $a\, b^m a \in L(A)$. By Lemma 7, $b^n \in L(A)$. Thus, $v\, b^k a\, b^m a\, b^n \in (L(A))^* L(A)\, L(A)\, L(A)$ $\subseteq$ {by definition of $(L(A))^*$} $\subseteq (L(A))^*$. □

THEOREM 11. $(a\, b^*\, a + b)^* = (a+b)^*_{ea}$.

PROOF. To show: Point (1): $(a\, b^*\, a + b)^* \subseteq (a+b)^*_{ea}$, and
Point (2): $(a+b)^*_{ea} \subseteq (a\, b^*\, a + b)^*$.

Point (1) follows from the fact that for all $n \geq 0$, $(a\, b^*\, a + b)^n \subseteq (a+b)^*_{ea}$. This can be shown by induction on n.

(*Basis*) $n=0$. Obvious, because ε is in $(a+b)^*$ and ε has an even number of a's.

(*Step*) Assume for n and show for $n+1$.

Consider a word $w \in (a\, b^*\, a + b)^{n+1}$. w belongs either to $(a\, b^*\, a + b)^n a\, b^*\, a$ or to $(a\, b^*\, a + b)^n b$. By induction hypothesis we know that every word in $(a\, b^*\, a + b)^n$ has an even number of a's. Since $a\, b^*\, a$ and b both have an even number of a's we get the thesis.

Point (2) is proved by induction on k, defined as the half of the number of a's in a word of $(a+b)^*_{ea}$.

(*Basis*) $k=0$. Obvious, because $b^* \subseteq (a\, b^*\, a + b)^*$.

(*Step*) Assume for k and show for $k+1$.

Consider a word $w \in (a+b)^*_{ea}$ such that w has $2(k{+}1)$ a's. w is of the form: $v\, a\, b^n\, a\, b^m$ for some $n, m \geq 0$ and some $v \in (a+b)^*_{ea}$ with $2k$ a's.

By inductive hypothesis $v \in (a\, b^*\, a + b)^*$. We also have that: $a\, b^n\, a \in a\, b^*\, a$ and $b^m \in b^*$.

Thus, we also have that: $v\,a\,b^n\,a\,b^m \in (a\,b^*\,a+b)^*\,(a\,b^*\,a)\,b^* \subseteq$
$\subseteq (a\,b^*\,a+b)^*\,(a\,b^*\,a+b)\,(a\,b^*\,a+b)^* \subseteq$
$\subseteq (a\,b^*\,a+b)^*$. □

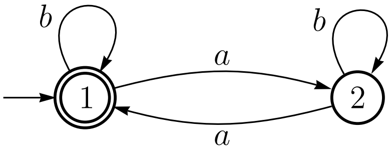

FIGURE 4. The finite automaton corresponding to the regular expression $(ab^*a+b)^*$. State 1 is the initial state and also a final state.

EXERCISE 12. Show that for any set S of words, we have that: $S^*\cdot S^* = S^*$.

Solution. ($\subseteq$). Take a word $w_i \in S^i$ and a word $w_j \in S^j$ for some $i, j \geq 0$. We have that $w_i \cdot w_j \in S^{i+j} \subseteq S^*$.
($\supseteq$). We have that $\{\varepsilon\}\cdot S^* = S^*$. □

7. Mathematical Induction (again) and Rule Induction

1. *Natural numbers.*

$N \rightarrow 0 \mid s\,N$

$L(N)$ denotes the set of words generated by the nonterminal N. Thus, $L(N)$ can be viewed as the set of the natural numbers written in unary notation (the number of s's tells us the natural number). Since we often denote the set of the natural numbers by the symbol N, the reader should not confuse between the nonterminal N and the set N of the natural numbers. Below the term $n+1$ stands for $s\,n$.

Rules for constructing $L(N)$: $\dfrac{\{\}}{0}$ $\qquad \forall n \in L(N).\ \dfrac{\{n\}}{s\,n}$

Rule Induction for $L(N)$ is given by the following two rules:

$$\frac{\mathit{true}}{P(0)} \qquad \forall n \in L(N).\ \frac{P(n)}{P(n+1)}$$

Also called Mathematical Induction or Peano Induction and represented as follows:

$$\frac{P(0) \qquad \forall n \in L(N).\ (P(n) \Rightarrow P(s\,n))}{\forall n \in L(N).\ P(n)}$$

2. Analogously, for any set of words (that is, a language) generated by a context-free grammar. For instance, given the grammar:

$A \rightarrow \varepsilon \mid A\,A \mid 0\,A\,1$

we have the following rules.

Rules for constructing $L(A)$:

$$\frac{\{\}}{\varepsilon} \qquad \forall a_1, a_2 \in L(A).\ \frac{\{a_1,\ a_2\}}{a_1\, a_2} \qquad \forall a \in L(A).\ \frac{\{a\}}{0\, a\, 1}$$

Rule Induction for $L(A)$ is given by the following three rules:

$$\frac{\mathit{true}}{P(\varepsilon)} \qquad \forall a_1, a_2 \in L(A).\ \frac{P(a_1) \quad P(a_2)}{P(a_1\, a_2)} \qquad \forall a \in L(A).\ \frac{P(a)}{P(0\, a\, 1)}$$

8. Equivalence of Context Free Grammars

Let us consider the context-free grammar with axiom A and the following productions:

$$A \ \to\ \varepsilon \ \mid\ A\,A \ \mid\ 0\,A\,1$$

Let us also consider the context-free grammar with axiom B' and the following productions:

$$B' \ \to\ 0\,B \qquad\qquad (\dagger 1)$$
$$B \ \to\ 1 \ \mid\ 0\,B\,B \qquad\qquad (\dagger 2)$$

Every word w generated by B' denotes a *mountain tour* of 0's and 1's. Every 0 denotes an uphill stretch and every 1 denotes an downhill stretch (see Figure 5). For every word w we have that: (i) every non-empty prefix of w has more 0's than 1's (that is, during the tour we never go below the altitude we started from), and
(ii) the number of 0's in w is equal to the number of 1's in w (that is, at the end of the tour, we return to the altitude we started from).

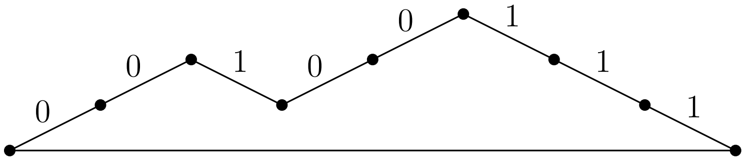

FIGURE 5. A word of B' is a mountain tour: 0 denotes an uphill stretch and 1 denotes downhill stretch.

For the productions (†1) and (†2) above, we have that following informal meaning:
(†1): «B' is a 0 followed by B», that is,
«B' is a 0 followed by a promise of an extra 1».
(†2): *either* B fulfils the promise of an 1 now *or* B generates one more 0 and makes two promises of an extra 1.

Here is the derivation of the word 001011 starting from B':

$$B' \to 0\,B \to 0\,0\,B\,B \to 0\,0\,1\,B \to 0\,0\,1\,0\,B\,B \to 0\,0\,1\,0\,1\,B \to 0\,0\,1\,0\,1\,1$$

Let $L(A)$, $L(B)$, and $L(B')$ denote the languages generated by the nonterminals A, B, and B', respectively.

Now we show by using Rule Induction that the languages generated by the axioms A and B' satisfy the following equality:

$$\boxed{L(A) = (L(B'))^*}$$

In order to do so, since $L(B') = 0\,L(B)$, it is enough to show that:

- *Point* (i): $L(A) \subseteq (0\,L(B))^*$ and
- *Point* (ii): $(0\,L(B))^* \subseteq L(A)$.

The rules for the language $L(A)$ are as follows:

$(A1)\ \dfrac{\{\}}{\varepsilon}$ $\qquad(A2)$ for any $a_1, a_2 \in L(A)$, $\dfrac{\{a_1,\ a_2\}}{a_1\,a_2}$ $\qquad(A3)$ for any $a \in L(A)$, $\dfrac{\{a\}}{0\,a\,1}$

The rules for the language $L(B)$ are as follows:

$(B1)\ \dfrac{\{\}}{1}$ $\qquad(B2)$ for any $b_1, b_2 \in L(B)$, $\dfrac{\{b_1,\ b_2\}}{0\,b_1\,b_2}$

Remark 13. We can use Rule Induction for proving the properties we will consider below because, as it is well known from the theory of formal languages, the inductive set of the rules $(A1)$, $(A2)$, and $(A3)$ is $L(A)$ and, analogously, the inductive set of the rules $(B1)$ and $(B2)$ is $L(B)$. □

For reasons of simplicity a singleton $\{x\}$ will be denoted also by x.

- *Proof of Point* (i).

It is enough to prove by Rule Induction that the property

$P(a) =_{def}\ a \in (0\,L(B))^*$

holds for any word $a \in L(A)$. This property is a consequence of the following Points (i.1), (i.2), and (i.3).

Point (i.1) By rule $(A1)$ we have to show that: $\varepsilon \in (0\,L(B))^*$. This is obvious.

Point (i.2) By rule $(A2)$ we have to show that: $a_1 a_2 \in (0\,L(B))^*$. Indeed, $a_1\,a_2 \in$ {by hypothesis} $\in (0\,L(B))^*(0\,L(B))^* =$ {by Exercise on page 9} $= (0\,L(B))^*$.

Point (i.3) By rule $(A3)$ we have to show that: $0\,a\,1 \in (0\,L(B))^*$. Indeed, $0\,a\,1 \in$ {by hypothesis} $\in 0\,L(A)\,1 \subseteq$ {by $B \to 1$} $\subseteq 0\,L(A)\,L(B) \subseteq$ {by Lemma 14 below} $\subseteq 0\,L(B) \subseteq (0\,L(B))^*$. □

- *Proof of Point* (ii).

It is enough to prove by Rule Induction that the property

$P(b) =_{def}\ (0\,b)^* \subseteq L(A)$

holds for any word $b \in L(B)$. This property is a consequence of the following Points (ii.1) and (ii.2).

Point (ii.1) By rule $(B1)$ we have to show that: $(0\,1)^* \subseteq L(A)$. This follows from $\forall n \geq 0$, $(01)^n \in L(A)$, which can be proved by induction on n.
(*Basis*) Since $(01)^0 = \varepsilon$ we have to show that $\varepsilon \in L(A)$. This holds because $A \to \varepsilon$.
(*Step*) Take any $k \geq 0$. We assume $(0\,1)^k \in L(A)$ and we show $(0\,1)^{k+1} \in L(A)$ as follows. We have that $(0\,1)^{k+1} = (0\,1)^k\,(0\,1) \in$ {by induction hypothesis} $\in L(A)\,0\,1 \subseteq$ {by $A \to 0\,A\,1 \to 0\,1$} $\subseteq L(A)\,L(A) \subseteq$ {by $A \to A\,A$} $\subseteq L(A)$.

Point (ii.2) By rule $(B2)$ we have to show that: $(0\,0\,b_1\,b_2)^* \subseteq L(A)$. Indeed, $(0\,0\,b_1\,b_2)^* \subseteq$ {by hypothesis} $\subseteq (0\,0\,L(B)\,L(B))^* \subseteq$ {by Lemma 15 below} $\subseteq (0\,L(A)\,L(B))^* \subseteq$ {by Lemma 14} $\subseteq (0\,L(B))^* \subseteq$ {by Lemma 15} $\subseteq (L(A))^* \subseteq L(A)$. This last inclusion holds because $\forall n.\,(L(A))^n \subseteq L(A)$, as it can be proved by induction on n as follows.

(*Basis*) The only word in $(L(A))^0$ is ε. Thus, we have to show that $\{\varepsilon\} \subseteq L(A)$. This holds because $A \to \varepsilon$.
(*Step*) Assume $(L(A))^n \subseteq L(A)$ and show $(L(A))^{n+1} \subseteq L(A)$. Now, $(L(A))^{n+1} = (L(A))^n\,L(A) \subseteq$ {by inductive hypothesis} $\subseteq L(A)\,L(A) \subseteq$ {by $A \to A\,A$} $\subseteq L(A)$.

This completes the proof of Point (ii.2) and the proof of the whole Point (ii). Thus, we have proved that $L(A) = (0\,L(B))^*$ (quomodo erat demonstrandum). □

Lemma 14. $L(A)\,L(B) \subseteq L(B)$.

Proof. We consider the property:

$Q(a) =_{def} a\,L(B) \subseteq L(B)$ and we show by Rule Induction that it holds for all $a \in L(A)$.

By rule $(A1)$ we have to show that: $\varepsilon\,L(B) \subseteq L(B)$. This is obvious.
By rule $(A2)$ we have to show that: $a_1\,a_2\,L(B) \subseteq L(B)$. Indeed, $a_1\,a_2\,L(B) \subseteq$ {by hypothesis} $\subseteq a_1\,L(B) \subseteq$ {by hypothesis} $\subseteq L(B)$.
By rule $(A3)$ we have to show that: $0\,a\,1\,L(B) \subseteq L(B)$. Indeed, $0\,a\,1\,L(B) \subseteq$ {by $B \to 1$} $\subseteq 0\,a\,L(B)\,L(B) \subseteq$ {by hypothesis} $\subseteq 0\,L(B)\,L(B) \subseteq$ {by $B \to 0\,B\,B$} $\subseteq L(B)$. □

Lemma 15. $0\,L(B) \subseteq L(A)$.

Proof. We consider the property:

$R(b) =_{def}\ 0\,b \in L(A)$ and we show by Rule Induction that it holds for all $b \in L(B)$.

By rule $(B1)$ we have to show that: $0\,1 \in L(A)$. This is obvious because: $A \to$ {by $A \to 0A1$} $\to 0A1 \to$ {by $A \to \varepsilon$} $\to 01$.
By rule $(B2)$ we have to show that: $0\,0\,b_1\,b_2 \in L(A)$. Indeed, $0\,0\,b_1\,b_2 \in$ {by hypothesis} $\in 0\,L(A)\,b_2 \subseteq$ {by hypothesis} $\subseteq 0\,L(A)\,L(B) \subseteq$ {by Lemma 16 below} $\subseteq 0\,L(A)\,L(A)\,1 \subseteq$ {by $A \to 0A1 \to 0AA1$} $\subseteq L(A)$. □

Lemma 16. $L(B) \subseteq L(A)\,1$.

Proof. We consider the property:

$S(b) =_{def}\ b \in L(A)\,1$ and we show by Rule Induction that it holds for all $b \in L(B)$.

By rule $(B1)$ we have to show that: $1 \in L(A)\,1$. This is obvious because $A \to \varepsilon$.
By rule $(B2)$ we have to show that: $0\,b_1\,b_2 \in L(A)\,1$. Indeed, $0\,b_1\,b_2 \in$ {by hypothesis} $\in 0\,L(A)\,1\,b_2 \subseteq$ {by $A \to 0A1$} $\subseteq L(A)\,b_2 \subseteq$ {by hypothesis} $\subseteq L(A)\,L(A)\,1 \subseteq$ {by $A \to AA$} $\subseteq L(A)\,1$. □

9. Regular Language Generated by a Context Free Grammar

We know that the problem *Reg*(G) of deciding whether or not a context-free grammar generates a regular language is undecidable. An instance of this problem is given by a context-free grammar G and the answer of an algorithm which would decide this problem should be "yes" if there exists a regular grammar equivalent to G and "no" if it does not exist a regular grammar equivalent to G. By the undecidability of *Reg*(G) we know that such an algorithm does not exist.

Since the problem $\neg \mathit{Reg}(G)$ of deciding whether or not a context-free grammar does *not* generate a regular language is semidecidable, we have, by Post Theorem, that the problem $\mathit{Reg}(G)$ is not even semidecidable.

In some cases, however, the problem $\mathit{Reg}(G)$ is semidecidable, and now we provide an algorithm for solving these semidecidable cases.

Let us consider a context-free grammar $G = \langle \Sigma, V_N, P, S \rangle$, where Σ is the set of the terminal symbols, V_N is the set of the nonterminal symbols, P is the set of productions, and S is the axiom. Let $L(G)$ denote the language generated the grammar G.

Let us consider the class of context-free grammars such that, for each $B \in V_N$, there exists a *regular expression* R_B which defines the language generated by B, denoted $L(B)$, as a function of the languages generated by the nonterminal symbols in V_N. Thus, the function R_B is an operator from (Powerset of $\Sigma^*)^{|V_N|}$ to Powerset of Σ^*. That operator is monotonic with respect to set inclusion, and since (Powerset of $\Sigma^*)^{|V_N|}$ is a lattice, by the Knaster-Tarski theorem, the function R_B has a minimal fixpoint and that minimal fixpoint is the language $L(B)$.

Then, we can construct a sequence $\langle L_0(S), L_1(S), \ldots \rangle$ of regular languages, such for all $i \geq 0$, $L_i(S) \subseteq L(G)$, and if $L(G)$ is a regular language, then there exists $k \geq 0$ such that $L_k(S) = L(G)$. The sequence $\langle L_0(S), L_1(S), \ldots \rangle$ can be constructed as we now indicate through an example.

Let us consider the grammar G with axiom S and the following productions:

$$\begin{array}{lcl} S & \rightarrow & A\,S \;\mid\; a \\ A & \rightarrow & S\,A \;\mid\; b \end{array}$$

We have that: $L(S) = L(A)^* a$ and $L(A) = L(S)^* b$ (these regular expressions for $L(S)$ and $L(A)$ can be derived by the Arden rule).

Then, we consider the following (possibly infinite) sequence of pairs of regular languages:

	0	1	...	i	...
for S :	$L_0(S)$	$L_1(S)$	...	$L_i(S)$	...
for A :	$L_0(A)$	$L_1(A)$	...	$L_i(A)$	...

$$\begin{array}{llcll} \text{where:} & \langle L_0(S), \;\; L_0(A) \rangle & = & \langle \emptyset, \;\; \emptyset \rangle & \text{and} \\ \text{for all } i \geq 0, & \langle L_{i+1}(S), \;\; L_{i+1}(A) \rangle & = & \langle L_i(A)^* a, \;\; L_i(S)^* b \rangle & \end{array}$$

If for some $k \geq 0$, we have that $L_{k+1}(S) = L_k(S)$, then we stop the construction of the sequence of pairs of languages for the nonterminals $\langle S, A \rangle$ and we get that $L(G) = L_k(S)$. (Recall that by Kleene's Theorem, the equivalence between two regular expressions can be tested by constructing the minimal finite automaton corresponding to each regular expression and then testing the isomorphism between these two minimal finite automata.) This result is a consequence of the fact that, for all nonterminal symbols $B \in V_N$, the language generated by B is the minimal fixpoint of the monotonic operator R_B.

In the case of the grammar G at hand we have:

	0	1	2	3	4
for S :	$\emptyset$	a	b^*a	$(a^*b)^*\,a$	$((b^*a)^*\,b)^*\,a$
for A :	$\emptyset$	b	a^*b	$(b^*a)^*\,b$	$((a^*b)^*\,a)^*\,b$

We need not proceed further in the construction of the sequence of languages for $\langle S, A\rangle$, because $(a^*b)^*\,a = ((b^*a)^*\,b)^*\,a$ and we have reached the minimal fixpoint.

Here is the proof of this fact. We make the proof by mathematical induction without referring to the minimal finite automata associated with the regular expression. In this proof, by abuse of language, we assume that a regular expression denotes both itself and the language it generates. We will also identify a singleton $\{a\}$ with the element a.

We have to show:
Point (i): $(a^*b)^*\,a \subseteq ((b^*a)^*\,b)^*\,a$, and
Point (ii): $(a^*b)^*\,a \supseteq ((b^*a)^*\,b)^*\,a$.

Point (i) is obvious, because $a^* \subseteq (b^*a)^*$ and inclusion of languages is preserved by the context $([\_]\,b)^*\,a$.

Point (ii) is a consequence of the fact that $(a^*b)^* \supseteq ((b^*a)^*\,b)^*$. This fact can be shown as follows.

$$\begin{aligned}
(a^*b)^* &= \{\text{by } e^* = (e^*)^*, \text{ for any regular expression } e\} = \\
&= ((a^*b)^*)^* = \{\text{by the following Lemma 17}\} = \\
&= ((a+b)^*b+\varepsilon)^* = \{\text{by definition of }^*\} = \\
&= ((a+b)^*b)^* = \{(a+b)^* \text{ is the set of all words over the alphabet } \{a,b\}\} = \\
&\supseteq ((b^*a)^*\,b)^*
\end{aligned}$$

Lemma 17. $(a^*b)^* = (a+b)^*b+\varepsilon$.

Proof. We have to show:
Point (i): $(a^*b)^* \subseteq (a+b)^*b+\varepsilon$, and
Point (ii) $(a+b)^*b+\varepsilon \subseteq (a^*b)^*$.
Point (i) follows from the facts that: (i.1) $(a+b)^*$ is the set of all words over the alphabet $\{a,b\}$, and (i.2) for all word $w \in (a^*b)^*$, either $w=\varepsilon$ or w ends with b.

Point (ii) follows from the facts that:

(ii.1) $\varepsilon \in (a^*b)^*$, and

(ii.2) for all $n \geq 0$, $(a+b)^n b \subseteq (a^*b)^+$ (and $(a^*b)^+ \subseteq (a^*b)^*$). Indeed,

(*Basis*) $n=0$. Obvious, because $b \in (a^*b)^+$.

(*Step*) Assume $(a+b)^n b \subseteq (a^*b)^+$. To show: $(a+b)^{n+1}b \subseteq (a^*b)^+$. Now,

$$\begin{aligned}
(a+b)^{n+1}b \;&= (a+b)\,(a+b)^n b = \\
&= a\,(a+b)^n b + b\,(a+b)^n b = \{\text{by induction hypothesis}\} = \\
&\subseteq a\,(a^*b)^+ + b\,(a^*b)^+.
\end{aligned}$$

We have that: (i) $a\,(a^*b)^+ \subseteq (a^*b)^+$, because any word in $a\,(a^*b)$ can be produced by (a^*b), and (ii) $b\,(a^*b)^+ \subseteq (a^*b)^+$, because $b \in (a^*b)$. □

10. Languages are Real Numbers and Real Numbers are Languages

Let us consider a non-empty finite set Σ of the terminal symbols.

We show that there is a bijection between the set of the subsets of Σ^*, denoted Powerset(Σ^*), and the set $\mathbb{R}_{(0,1)}$ of the real numbers in the open interval $(0,1)$. This fact can be expressed as follows:

$$|\,\mathrm{Powerset}(\Sigma^*)\,| \;=\; |\,\mathbb{R}_{(0,1)}\,|. \qquad (\ddagger)$$

Recall that: (i) Powerset(Σ^*) is the set of languages over Σ, and (ii) there is a bijection between the set $\mathbb{R}_{(0,1)}$ and the set $\mathbb{R}_{(-\infty,+\infty)}$ of the real numbers in the interval $(-\infty,+\infty)$ (for this Point (ii) see also Point (i) at the beginning of Section 12 on page 19).

NOTATION. Let 0^ω and 1^ω denote the *infinite* string of 0's and 1's, respectively. Analogously, for any digit d, let d^ω denote the infinite string of d's. □

By definition of a real number, we assume that any real number $x \in \mathbb{R}_{(0,1)}$ whose infinite binary expansion is $0.v\,1\,0^\omega$, for some $v \in \{0,1\}^*$, is identified with the real number whose infinite binary expansion is $0.v\,0\,1^\omega$.

For instance, the real number with binary expansion 0.10110^ω is the same real number with binary expansion 0.10101^ω (obtained from 0.10110^ω by replacing the rightmost substring 10^ω by 01^ω). That real number is 0.6875 (in decimal notation) because $0.6875 = 0.5+0.125+0.0625 = 2^{-1}+2^{-3}+2^{-4} = 0.10110^\omega$.

Unless otherwise stated, we assume that the infinite binary expansion of any real number is the one that does *not* end with 0^ω to its right.

Note that also in the decimal notation we have a similar identification. For instance, the infinite sequences 6.0^ω and 5.9^ω both denote the same real number 6.

Let us introduce the following definitions.

DEFINITION 18. [**Fractional Binary Expansion $b(x)$ of a Real Number x in $\mathbb{R}_{(0,1)}$**] Given any real number $x \in \mathbb{R}_{(0,1)}$, by $b(x)$ we denote the infinite sequence w such that $0.w$ is the *unique* infinite binary expansion of x which does *not* end with 0^ω to its right (thus, $b(x)$ has an infinite number of 1's).

For instance, we have that $b(0.6875) = 10101^\omega$.

DEFINITION 19. [**Position in a Sequence**] Given a sequence of 0's and 1's, we say that its leftmost bit is in *position* 0 and we say that the bit immediately to the right of the bit in position p is in position $p+1$.

Given a sequence σ in $\{0,1\}^\omega$, for all $i \geq 0$, by $\sigma[i]$ we denote the bit in position i.

- *Proof of the existence of the bijection stated by* ($\ddagger$) *above.*

Let N denote the set of the natural numbers. The proof of the bijection stated by ($\ddagger$) above follows from the existence of:

(i) a bijection α between the Powerset(N) and the set $\mathbb{R}_{(0,1)}$, and

(ii) a bijection β between N and Σ^*, because (as the reader may easily verify):

(i) for any two sets A and B, if there is a bijection between A and B, then there is a bijection between Powerset(A) and Powerset(B), and

(ii) the composition of two bijections is a bijection.

Definition of the bijection α from Powerset(N) *to* $\mathbb{R}_{(0,1)}$.

The existence of the bijection α is a consequence of the Bernstein Theorem and the following two injections r from Powerset(N) to $\mathbb{R}_{(0,1)}$ and s from $\mathbb{R}_{(0,1)}$ to Powerset(N).

Here is the definition of the injection r: for all $A \subseteq N$,
if A is infinite, then $r(A) = x$ such that:

(i) $b(x)[0] = 1$,
(ii) $b(x)[1] = 0$, and
(iii) for all $i \geq 0$, $b(x)[i+2] = 1$ iff $i \in A$

if A is finite with maximal element max, then $r(A) = x$ such that:

(i) $b(x)[0] = 0$,
(ii) $b(x)[1] = 1$,
(iii) for all i, with $0 \leq i < max$, $b(x)[i+2] = 1$ iff $i \in A$,
(iv) $b(x)[max+2] = 0$, and
(v) for all i, with $i > max$, $b(x)[i+2] = 1$

if $A = \emptyset$, then $r(A) = x$ such that:

(i) $b(x)[0] = b(x)[1] = 0$, and
(ii) for all $i \geq 0$, with $b(x)[i+2] = 1$

Thus, any infinite subset of N is mapped to a real in the semi-open interval (0.500, 0.750] and the set N is mapped to 0.750. Every finite subset of N is mapped to a real in the semi-open interval [0.250, 0.500) and the empty set $\emptyset$ is mapped to 0.250.

Remark 20. The definition of the injection r *cannot* be as follows:

for all $A \subseteq N$, $r(A) = x$ such that for all $i \geq 0$, $b(x)[i] = 1$ iff $i \in A$

because we identify the real number denoted by $0.v\,1\,0^\omega$ with the real number denoted by $0.v\,0\,1^\omega$, for any $v \in \{0,1\}^*$. Thus, by this definition of r, we would have that two distinct subsets of N are mapped to the same real number.

Obviously, one could define an injection from Powerset(N) to $\mathbb{R}_{(0,1)}$ which is different from r. We leave it as an exercise to the reader to give the definition of such an injection different from r.

The injection s is defined as follows: for all $x \in \mathbb{R}_{(0,1)}$,

$s(x) =_{def} \{i \mid b(x)[i] = 1\}$.

For all $x \in \mathbb{R}_{(0,1)}$, since $b(x)$ has an infinite number of 1's, $s(x)$ is an infinite subset of N.

Definition of the bijection β from N to Σ^*.

The existence of the bijection β is a consequence of the Bernstein Theorem and the following two injections f from N to Σ^* and g from Σ^* to N.

First, we define the injection f. Let a be an element of Σ. Then, f is defined as follows: for every $n \in N$, $f(n) = a^n$. In particular, $f(0) = \varepsilon$.

Then, we define the injection g from Σ^* to N. Let us assume, without loss of generality, that $\Sigma = \{a, b\}$. Let us consider the elements of Σ^* in their *canonical order*. According to this order: (i) shorter words are listed before longer words, and (ii) words of equal length are listed in the lexicographic order, whereby, for instance, $a < b$, $a < aa$, and $aa < ab$. Thus, we have that all the words of Σ^* are listed as follows:

$\langle \varepsilon, a, b, aa, ab, ba, bb, aaa, aab, aba, abb, baa, \ldots \rangle$

We define the injection g as the following table shows:

w :	ε	a	b	aa	ab	ba	bb	aaa	aab	aba	abb	baa	bab	...
$g(w)$:	0	1	2	3	4	5	6	7	8	9	10	11	12	...

Remark 21. It is easy to see that, for all $w \in \{a, b\}^*$,

$g(w) = (1w[a/0,\, b/1])_2 - 1 \in N$

where: (i) $w[a/0, b/1]$ denotes the string w with 0 and 1, instead of a and b, respectively, (ii) for all $v \in \{0, 1\}^*$, $(v)_2$ denotes the natural number whose binary expansion is v.

We have that g is actually a *bijection* between Σ^* and N, and thus we can take β to be g. The easy proof of this fact is left to the reader.

• *Alternative proof of the existence of the bijection between* Powerset(Σ^*) *and* $\mathbb{R}_{(0,1)}$ *stated by* (‡) *above.*

This proof is based on the Bernstein Theorem and the following two injections: (i) the injection h from Powerset(Σ^*) to $\mathbb{R}_{(0,1)}$, and (ii) the injection k from $\mathbb{R}_{(0,1)}$ to Powerset(Σ^*), that we now define.

Definition of the injection h from Powerset(Σ^*) *to* $\mathbb{R}_{(0,1)}$.

Without loss of generality, let us consider the alphabet $\Sigma = \{a, b\}$. Let us also define an encoding σ of the symbols in Σ as follows: $\sigma(a) = 00$ and $\sigma(b) = 01$.

Let us also consider a language $L \subseteq \Sigma^*$ presented as a sequence Λ of words in the canonical order. Thus, every non-empty word w in L is encoded by sequence $\sigma(w) \in \{00, 01\}^+$. We assume that the encoding of the symbol comma ',' separating any two words in the presentation Λ of L is 11, and the encoding of the empty sequence $\varepsilon \in \Sigma^*$ is the sequence 10.

Now we define the injection h mapping $\Lambda = \langle w_0, w_1, \ldots\rangle$ to a sequence of 0's and 1's which should be interpreted as fractional binary expansion $b(x)$ of a real number x in $\mathbb{R}_{(0,1)}$. The injection h is defined as follows:

if Λ is infinite, then $h(\Lambda) = \sigma(w_0)\,\gamma\,\sigma(w_1)\,\gamma \ldots$,

if Λ is finite, say $\Lambda = \langle w_0, w_1, \ldots, w_n\rangle$, then $h(\Lambda) = \sigma(w_0)\,\gamma\,\sigma(w_1)\,\gamma \ldots \gamma\,\sigma(w_n)\,1^\omega$.

Note that: (i) $h(\langle\rangle) = 1^\omega$, (ii) $h(\langle\varepsilon\rangle) = 101^\omega$, and (iii) if L is infinite, then $h(\Lambda)$ does not end with all 0's to the right (recall that comma is encoded by 11).

The reader may easily show that, since every language in Σ^* is presented in the canonical order, h is an injection from Powerset(Σ^*) to $\mathbb{R}_{(0,1)}$.

Definition of the injection k from $\mathbb{R}_{(0,1)}$ *to* Powerset(Σ^*).

Given any real number x in $\mathbb{R}_{(0,1)}$, we stipulate that:

$k(x) = \{a^i \mid i \geq 0 \text{ and } b(x)[i] = 1\}$.

It is easy to see that k is indeed an injection from $\mathbb{R}_{(0,1)}$ to Powerset(Σ^*). □

11. Counting Formulas and n-ary Trees

• *Counting Formulas.* The set $\mathcal{F}$ of formulas of the Propositional Calculus is defined as follows, starting from the infinite set $\mathcal{P} = \{P_i \mid i \in N\}$ of *elementary propositions.*

$\mathcal{F} \ni \varphi \qquad\qquad \varphi \ ::= \ P_0 \mid P_1 \mid \ldots \mid \neg\,\varphi \mid (\varphi \to \varphi)$

Obviously, $|\mathcal{P}| = |N|$.

Now we show that $|\mathcal{F}| = |N|$. This equality is a consequence of the Bernstein Theorem and the existence of the following two injections: (i) f from $\mathcal{F}$ to N, and (ii) g from N to $\mathcal{F}$.

The injection f is defined as follows. First, we encode any formula φ by a sequence $\sigma(\varphi)$ of digits in base 6 obtained by replacing:

(i) for all $i \in N$, the proposition P_i by the binary expansion of i in $\{0,1\}^+$,
(ii) '$\neg$' by 2,
(iii) '$\rightarrow$' by 3,
(iv) '(' by 4, and
(v) ')' by 5.

Thus, P_0, P_1, P_2, $P_3, \ldots$ are encoded by 0, 1, 10, 11, ..., respectively. For instance, $P_3 \rightarrow (\neg P_2 \rightarrow P_3)$ is encoded by the sequence 11342103115.

For all $\varphi \in \mathcal{F}$, we define $f(\varphi)$ to be the natural number whose expansion in base 6 is $\sigma(\varphi)$.

The injection g is defined as follows: for all $i \in N$, $g(i) = P_i$.

• *Counting n-ary Trees.* The set $\mathcal{T}$ of n-ary trees whose leaves and internal nodes have values taken from a non-empty set A such that $|A| = |N|$, is defined by the following domain equations:

$$\mathcal{T} \;=\; \{\bullet\} + A + A \times \mathit{List}(\mathcal{T}) \qquad (1)$$

$$\mathit{List}(\mathcal{T}) \;=\; \{[\,]\} + \mathcal{T} \times \mathit{List}(\mathcal{T}) \qquad (2)$$

where '$\bullet$' denotes the empty tree and '[ ]' denotes the empty list of n-ary trees in $\mathcal{T}$. Note that the empty tree $\bullet$ is different from the empty list [ ] of n-ary trees. As usual, we denote a list using the square brackets notation. For instance, $[b,\ a,\ b,\ c]$ denotes the list whose elements are: b, a, b, and c, in this order.

In Equation (1) the summand A denotes the set of n-ary trees which are leaves with values in A. The summand $A \times \mathit{List}(\mathcal{T})$ denotes the set of internal nodes whose values are in A and whose child nodes are represented 'from left to right' as a list of n-ary trees in $\mathcal{T}$. Thus, for instance, the list $[a,\ [t_1, \ldots, t_n]]$ denotes an internal node with value $a \in A$ and whose child nodes are the n trees $t_1, \ldots, t_n$, from left to right, in this order. (Note that here and in what follows we equivalently use the word 'node' and the word 'tree'.)

Equation (2) denotes, as usual, the set of lists whose elements are n-ary trees.

Figure 6 depicts the n-ary tree $[b,\ [\bullet,\ [b,\ [c,\ a]],\ a]]$.

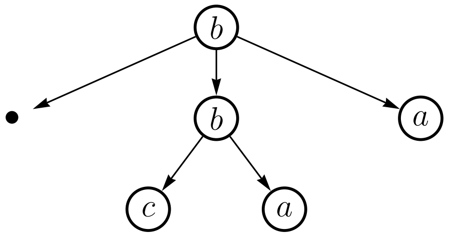

FIGURE 6. The n-ary tree $[b,\ [\bullet,\ [b,\ [c, a]],\ a]]$.

Note that in the domain of n-ary trees we distinguish between: (i) a leaf node with value a, and (ii) a node with value a having an empty list of child nodes (even if we may decide not to distinguish between them in their pictorial representations).

REMARK 22. The n-ary trees we consider here are: (i) *oriented*, in the sense that every node has an directed arc from the node itself to any of its child nodes, and (ii) *ordered*, in the sense that the child nodes are ordered in a list 'from left to right'. □

Now we show that $|\mathcal{T}| = |N|$. This equality is a consequence of the Bernstein Theorem and the existence of the following two injections: (i) f from $\mathcal{T}$ to N, and (ii) g from N to $\mathcal{T}$.

The injection f is defined as follows. First, we encode any n-ary tree t by a sequence $\sigma(t)$ of digits in base 6 obtained by replacing:

(i) every element of A by a suitable binary encoding in $\{0,1\}^+$,
(ii) '•' by 2,
(iii) ',' by 3,
(iv) '[' by 4, and
(v) ']' by 5.

For instance, if we assume that the elements a, b, and c are encoded by the binary strings 00, 01, and 10, respectively, then the n-ary tree $[b,\ [\bullet,\ [b,\ [c,a]],\ a]]$ is encoded by the sequence 40134234013410300553005.

For all n-ary trees $t \in \mathcal{T}$, we define $f(t)$ to be the natural number whose expansion in base 6 is $\sigma(t)$.

The injection g is defined as follows. Let a be an element of A. For all $n \in N$, $g(n) = [a,\ [a, a, \ldots, a]]$, that is, a n-ary tree whose root has the value a and whose n child nodes are leaves with the same value a (see Figure 7). For $n=0$, we have that $g(0)=[a,\ [\,]]$.

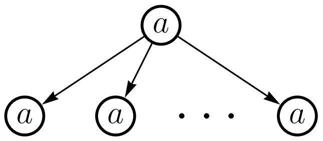

FIGURE 7. A n-ary tree whose root has value a and whose n (≥ 0) child nodes are all leaves with value a.

12. Some Properties of Real Numbers and Languages

Let us consider a finite alphabet $\Sigma = \{0,1\}$. Let Σ^* be the set of all *finite* sequences of 0's and 1's. Let Σ^ω be the set of all *infinite* sequences of 0's and 1's. We have that:

(i) there exists a bijection between the set $\mathbb{R}_{(-\infty,+\infty)}$ of the real numbers in the interval $(-\infty,+\infty)$ and the set $\mathbb{R}_{(0,1)}$ of the real numbers in the open interval $(0,1)$,

(ii) there exists a bijection between $\mathbb{R}_{(0,1)}$ of the real numbers in $(0,1)$ and the set Σ^ω of the infinite strings of 0's and 1's, and

(iii) there exists a bijection between Powerset(Σ^*), that is, the set of the (finite or infinite) subsets of Σ^* (in other words, the set of languages over Σ), and the set Σ^ω.

• For Point (i) we have that $\lambda x.\frac{2\,\mathrm{arctg}(e^x)}{\pi}$ is a bijection from $\mathbb{R}_{(-\infty,+\infty)}$ to $\mathbb{R}_{(0,1)}$.

• For Point (ii) one can use the Bernstein Theorem and the two injections f from $\mathbb{R}_{(0,1)}$ to Σ^ω and g from Σ^ω to $\mathbb{R}_{(0,1)}$ we now define.

First, recall that, by definition, the set $\mathbb{R}_{(0,1)}$ can be viewed as the set $\{0,1\}^\omega - (\{0^\omega, 1^\omega\} \cup \{0,1\}^* 10^\omega)$ of the fractional binary expansions (indeed, 0 and 1 are not in $\mathbb{R}_{(0,1)}$ and, for any $v \in \{0,1\}^*$, we write $v01^\omega$ instead of $v10^\omega$).

The definition of the injection f is based on the fact that every real number in $\mathbb{R}_{(0,1)}$ has a unique infinite binary expansion in Σ^ω not ending with all 0's (see Definition 18 on page 15).

The definition of the injection g is as follows. For any given infinite sequence w in Σ^ω, we first construct the list P of its non-empty prefixes in lexicographic order. If $P = \langle p_0, p_1, p_2, p_3, \ldots \rangle$, then $g(w)$ is the real number $x \in \mathbb{R}_{(0,1)}$ such that $b(x)$ (see again Definition 18 on page 15) is given by the concatenation $p_0 0 p_1 1 p_2 0 p_3 1 \ldots$ (since the separating bits are 0 and 1, this concatenation cannot end with 0^ω.) For instance, for the sequence 1^ω, we have that P is $\langle 1, 11, 111, 1111, \ldots \rangle$, and thus the real number $x = g(1^\omega)$ is (in binary) $0.1\,0\,11\,1\,111\,0\,1111\,1\ldots$.

Remark 23. A different definition of an injection g from Σ^ω to $\mathbb{R}_{(0,1)}$ is as follows. For any given infinite sequence w in Σ^ω, we stipulate that:

$$g(0w) = 001\,g(w)$$

$$g(1w) = 101\,g(w)$$

Note that for any w in Σ^ω, we have that $g(w)$ does not end with 0^ω.

• For Point (iii) one can use the Bernstein Theorem and the two injections from Powerset(Σ^*) to Σ^ω and from Σ^ω to Powerset(Σ^*) we now define.

An injection from Powerset(Σ^*) to Σ^ω can be constructed as we now illustrate through an example.

Consider the language $A = \{01,\ 0101,\ 010101, \ldots\}$, that is, $\{(01)^n \mid n \geq 1\}$ presented as the list $\Lambda(A) = \langle 01,\ 0101,\ 010101, \ldots \rangle$ of words in the canonical order. The language A is mapped to the infinite sequence $\sigma(A)$ constructed by encoding every word and comma in $\Lambda(A)$ as indicated by the following table:

$$\begin{array}{rcccccccl}
A = \{ & 01 & , & 0101 & , & 010101 & , & \ldots & \} \\
\Lambda(A) = \{ & 01 & , & 0101 & , & 010101 & , & \ldots & \rangle \\
\sigma(A) = & 0001 & 11 & 00010001 & 11 & 000100010001 & 11 & \ldots &
\end{array}$$

where: (i) $\sigma(0)=00$, (ii) $\sigma(1)=01$, and (iii) $\sigma(,)=11$. (For readability reasons we have inserted in the sequence $\sigma(A)$ blank characters to the left and to the right of the 11's which encode commas.)

We assume that σ maps the empty word ε to 10. If the language is a *finite* subset of Σ^*, then $\sigma(A)$ is obtained by first encoding every word and comma in $\Lambda(A)$ as indicated in the table above, and then concatenating the sequence 1^ω to the right. For instance, the empty language is mapped to 1^ω, and the language $\{\varepsilon, 0\}$ is mapped to 1011001^ω.

It is not difficult to show that the function σ is an injection from Powerset(Σ^*) to Σ^ω.

An injection from Σ^ω to Powerset(Σ^*) can be constructed by considering, for any given infinite sequence in Σ^ω, the infinite set of its prefixes. This set is a language subset of Σ^*.

13. Some Properties of Turing Machines

For each recursive enumerable language $A \subseteq \Sigma^*$, there exists a Turing Machine M such that A is recognized by M, that is, $w \in A$ iff M stops and accepts w. In this case we write: $L(M) = A$. Actually, if there exists a Turing Machine M which recognizes A, then there exist $\aleph_0$ Turing Machines, each of which recognizes A.

Recall that every Turing Machine M has an associated binary code m which is a word in Σ^*.

• Given a recursive enumerable language $A \subseteq \Sigma^*$, we may define the following infinite language $M_A \subseteq \Sigma^*$ of Turing Machine codes:

$$M_A = \{m \mid m \text{ is the code of a Turing Machine } M \text{ such that } L(M) = A\}.$$

The cardinality of M_A is $\aleph_0$.

• Given a *set S of recursively enumerable languages,* we may define the following infinite language $M_S \subseteq \Sigma^*$ of Turing Machine codes:

$$M_S = \{m \mid m \text{ is the code of a Turing Machine } M \text{ such that } L(M) = A \text{ and } A \in S\}.$$

The cardinality of M_S is $\aleph_0$.

• We have that M_S is *recursively enumerable* for the following sets S of languages, for any $k \geq 0$, for any $w \in \Sigma^*$:

$S = \{A \mid A \text{ is not empty}\}$

(that is, there is a Turing Machine which enumerates the codes of *all* the Turing Machines each of which recognizes a non-empty language),

$S = \{A \mid A \text{ has at least } k \text{ words}\}$, and

$S = \{A \mid w \in A\}$.

• We have that M_S is *not recursively enumerable* for the following sets S of languages:

$S = \{\emptyset\}$ (that is, there is no Turing Machine which recognizes the code of *all* the Turing Machines each of which recognizes the empty language, that is, each of which accepts no word),

$S = \{\Sigma^*\}$,

$S = \{A \mid A \text{ is recursive}\}$

(that is, there is no Turing Machine which recognizes the code of *all* the Turing Machines which always terminate),

$S = \{A \mid A \text{ is context-sensitive}\}$,

$S = \{A \mid A \text{ is context-free}\}$, and

$S = \{A \mid A \text{ is regular}\}$

(that is, there is no Turing Machine which recognizes the code of *all* the Turing Machines which recognize a regular language).

Thus, for instance, it is semidecidable whether or not a given Turing Machine accepts a non-empty language, while it is not semidecidable whether or not a given Turing Machine accepts no words at all, that is, it accepts the empty language.

Note that whatever we said in this section is parameterized by the choice of the alphabet Σ. For instance, we could have taken Σ to be the set $\{a, b\}$ or the set $\{a, b, c\}$, instead of the set $\{0, 1\}$.