substitution

Canonical name	Substitution
Date of creation	2013-03-22 18:55:12
Last modified on	2013-03-22 18:55:12
Owner	CWoo (3771)
Last modified by	CWoo (3771)
Numerical id	20
Author	CWoo (3771)
Entry type	Definition
Classification	msc 68Q45
Synonym	string substitution
Related topic	HomomorphismOfLanguages
Related topic	SubstitutionsInLogic
Defines	λ-free substitution
Defines	regular substitution

Definition

Let Σ_1, Σ_2 be alphabets. A *substitution*, or *string substitution*, is a function $s : \Sigma_1^* \to P(\Sigma_2^*)$ such that

- s preserves the empty word: $s(\lambda) = \{\lambda\}$, and
- s preserves concatenation: $s(\alpha\beta) = s(\alpha)s(\beta)$.

In other words, for every word α over Σ_1, $s(\alpha)$ is a language over Σ_2. In the second condition above, $s(\alpha)s(\beta)$ is the concatenation of languages: $\{uv \mid u \in s(\alpha), v \in s(\beta)\}$.

For example, suppose $\Sigma = \{a, b\}$. The map s taking u to $\{u'\}$, where u' is obtained from u by replacing every occurrence of a by b is a substitution.

One easy way to obtain more examples of substitutions is to start with some function

$$f : \Sigma_1 \to P(\Sigma_2^*),$$

and extend it to all of Σ_1^* by language concatenation: if $u = a_1 \cdots a_n$, with $a_i \in \Sigma_1$, defining

$$s(u) := f(a_1) \cdots f(a_n)$$

gives us a substitution s. It is easy to see that the extension is unique (if s_1 and s_2 both extend f, then $s_1 = s_2$).

In fact, every substitution is obtained this way: every substitution $s : \Sigma_1^* \to P(\Sigma_2^*)$ is the extension of its restriction to Σ_1. This can be verified directly, but is the result of a more general fact: any function $f : A \to B$, where B is a semigroup, extends uniquely to a semigroup homomorphism $f^* : A^* \to B$ where A^* is the semigroup freely generated by A.

In the previous example, s is the extension of the function that takes a to $\{b\}$ and b to $\{b\}$.

Closure under Substitution

For any language $L \subseteq \Sigma_1^*$ and a substitution $s : \Sigma_1^* \to P(\Sigma_2^*)$, define

$$s(L) := \bigcup \{s(u) \mid u \in L\}.$$

A family $\mathscr{F}$ of languages is said to be *closed under substitutions* if, given any substitution s, with $L \in \mathscr{F}$ and $s(w) \in \mathscr{F}$ for each $w \in L$, we have $s(L) \in \mathscr{F}$. The following families are closed under substitutions:

- regular languages,
- context-free languages, and
- type-0 langauges.

As a corollary, the families of regular, context-free, and type-0 languages are closed under homomorphisms, since every homomorphism of languages is really just a special case of substitution, such that every symbol of the domain alphabet is mapped to a singleton consisting of a word over the range alphabet.

The family of context-sensitive languages is not closed under general substitutions. Instead, it is closed under λ-free substitutions (see remark below).

Remarks.

- The notion of string substitution generally corresponds to our intuitive notion of how a substitution should behave:

  > given words u, v, w, then Substitute(u, v, w) is a word that is obtained from u by replacing every occurrence of v in u by w.

  However, this is not always the case. For example, let $\Sigma = \{a, b\}$, and s be the map that takes u to $\{u'\}$, where u' is obtained from u by replacing all occurrences of aa, if any, by b. Then it is easy to see that s is not a substitution, for

  $$s(a)s(a) = \{a\}\{a\} = \{aa\}$$

  while

  $$s(aa) = \{b\} \neq s(a)s(a).$$

  Nevertheless, s is "intuitively" a "substitution".

- A substitution s is said to have property $\mathcal{P}$ if for each $a \in \Sigma$, the set $s(a)$ has property $\mathcal{P}$. Thus, for example, a substitution s is finite if $s(a)$ is a finite set, regular if $s(a)$ is a regular language, and λ-free if each $s(a)$ is λ-free, etc...

References

[1] S. Ginsburg, *The Mathematical Theory of Context-Free Languages*, McGraw-Hill, New York (1966).

[2] D. C. Kozen, *Automata and Computability*, Springer, New York (1997).