Seven families of language equations

Alexander Okhotin[*]

22 May 2007

Equations with formal languages as unknowns are among the most natural objects of study in language theory. The research of their properties dates back to a paper by Ginsburg and Rice [1962], in which the semantics of the context-free grammars was equivalently defined using systems of equations of the basic form

$$\left.\begin{aligned} X_1 &= \varphi_1(X_1, \ldots, X_n) \\ &\;\;\vdots \\ X_n &= \varphi_n(X_1, \ldots, X_n) \end{aligned}\right\} \tag{*}$$

where each φ_i contains variables and singleton constant languages connected with union and concatenation.

For example, consider the one-variable equation $X = (\{a\} \cdot X \cdot \{b\}) \cup \{\varepsilon\}$ over the alphabet $\Sigma = \{a, b\}$, denoted $X = aXb \cup \varepsilon$ for short. It represents the context-free grammar $S \to aSb \mid \varepsilon$ and its unique solution is $\{a^n b^n \mid n \geqslant 0\}$. Another example is the equation $X = XX \cup aXb \cup \varepsilon$ with multiple solutions, of which the least solution is the Dyck language and the greatest solution is Σ^*.

After some years of not more than occasional study, the subject of language equations has recently undergone a revival. Equations of various form have been investigated, and their different properties were established. In particular, language equations with non-standard word operations were considered by Kari [1994], and her approach received a considerable following. The equation $XL = LX$, where L is a regular constant, has been studied by several authors, until, contrary to the conjectures, Kunc [2005] established its computational universality. Equations and inequalities with concatenations and Boolean operations were studied by Okhotin [2002, 2003, 2005b, 2006]. A recent survey by Kunc [2007] summarizes the existing results in the area.

Enough knowledge has been accumulated to attempt classification of the known types of language equations. This paper begins such a classification by considering systems of the above form (*), in which any fixed set of Boolean operations can be used instead of $\{\cup\}$. As we know from Ginsburg and Rice [1962], for union only these systems represent the context-free languages. A few other cases have been considered in the recent years. It is left to summarize this knowledge and to determine the exact number of families of languages generated by unique solutions of such equations. An essential tool for this study is the fundamental work by Post [1941] on the classes of Boolean functions.

Post's lattice

In his seminal paper, Post [1941] completely described all classes of Boolean functions closed under superposition: there are 8 countable hierarchies and 44 individual classes organized into an elaborate lattice of containment. Each class has a finite basis. A clear proof and explanation of Post's results can be found in a textbook by Yablonski et al. [1966].

Post refers to his classes as "closed systems", while subsequent literature has coined the term "clone". For every set of Boolean functions $\mathcal{F}$, denote its closure under superposition by $[\mathcal{F}]$. We shall use the following symbols for Boolean functions: $\wedge$ for conjunction, $\vee$ for disjunction, $\neg$ for negation and $\oplus$ for sum modulo 2 (exclusive or).

Post's lattice, written in the notation of Yablonski et al. [1966], is shown in the bottom layer of Figure 1. The class P_2 at the top of the figure is the class of all Boolean functions, generated, for instance, as $[x \vee y, \neg x]$, and each of the rest of the families has its own basis, such as $D_{01} = [x \vee y]$. Each line specifies a proper inclusion.

Let us apply Post's lattice to language equations. We shall consider finite sets of Boolean functions $\mathcal{F}$, and interpret them as set-theoretic functions operating on subsets of Σ^*. Then logical constant 0 represents $\varnothing$, constant 1 represents Σ^*, disjunction represents union, sum modulo two represents symmetric difference, etc.

For every fixed set of Boolean functions $\mathcal{F}$, consider systems (*) with concatenation, Boolean operations from $\mathcal{F}$ and singleton constants, which have a unique solution. Such a system is regarded as a specification of languages that are components of its unique solution. Hence, every set of Boolean functions generates a family of formal languages.

Since set-theoretic operations can be freely combined in the right-hand sides of equations (*), given a basis set of operations $\mathcal{F}$, one can directly express any Boolean function in $[\mathcal{F}]$. Thus we have to consider countably many Post's classes and can use Post's complete description of these classes to describe the corresponding language equations.

The main result of this paper is that seven distinct families are obtained in this way. These families are denoted O, I, K, D, M, N and P, more or less after their respective generating classes of Boolean functions. These families are described in the rest of this paper.

D: Disjunction only

If the only allowed Boolean operation is disjunction, one obtains the well-known equations of Ginsburg and Rice [1962], which generate the family of the context-free languages. Better to say, these equations provide a natural semantics to the context-free grammars, which is equivalent to Chomsky's operational semantics by derivation.

These equations normally use the basis of Boolean functions $\{x \vee y\}$. Adding constants to this basis clearly does not increase their expressive power, hence the following theorem is obtained:

Theorem 1. *Let $\mathcal{F}$ be a class of Boolean functions, such that $[x \vee y] \subseteq [\mathcal{F}] \subseteq [x \vee y, 0, 1]$. Then unique solutions of systems (*) with Boolean operations from $\mathcal{F}$, concatenation and singleton constants define the context-free languages.*

There are four closed classes of Boolean functions satisfying these conditions, namely, $D_{01} = [\{x \vee y\}]$, $D = [\{x \vee y, 0, 1\}]$, $D_0 = [\{x \vee y, 0\}]$ and $D_1 = [\{x \vee y, 1\}]$. By the theorem, each of them yields the family of context-free languages.

The term "context-free languages" comes from string rewriting. In our framework, this is the family of languages generated by the disjunction, and it will accordingly be called D.
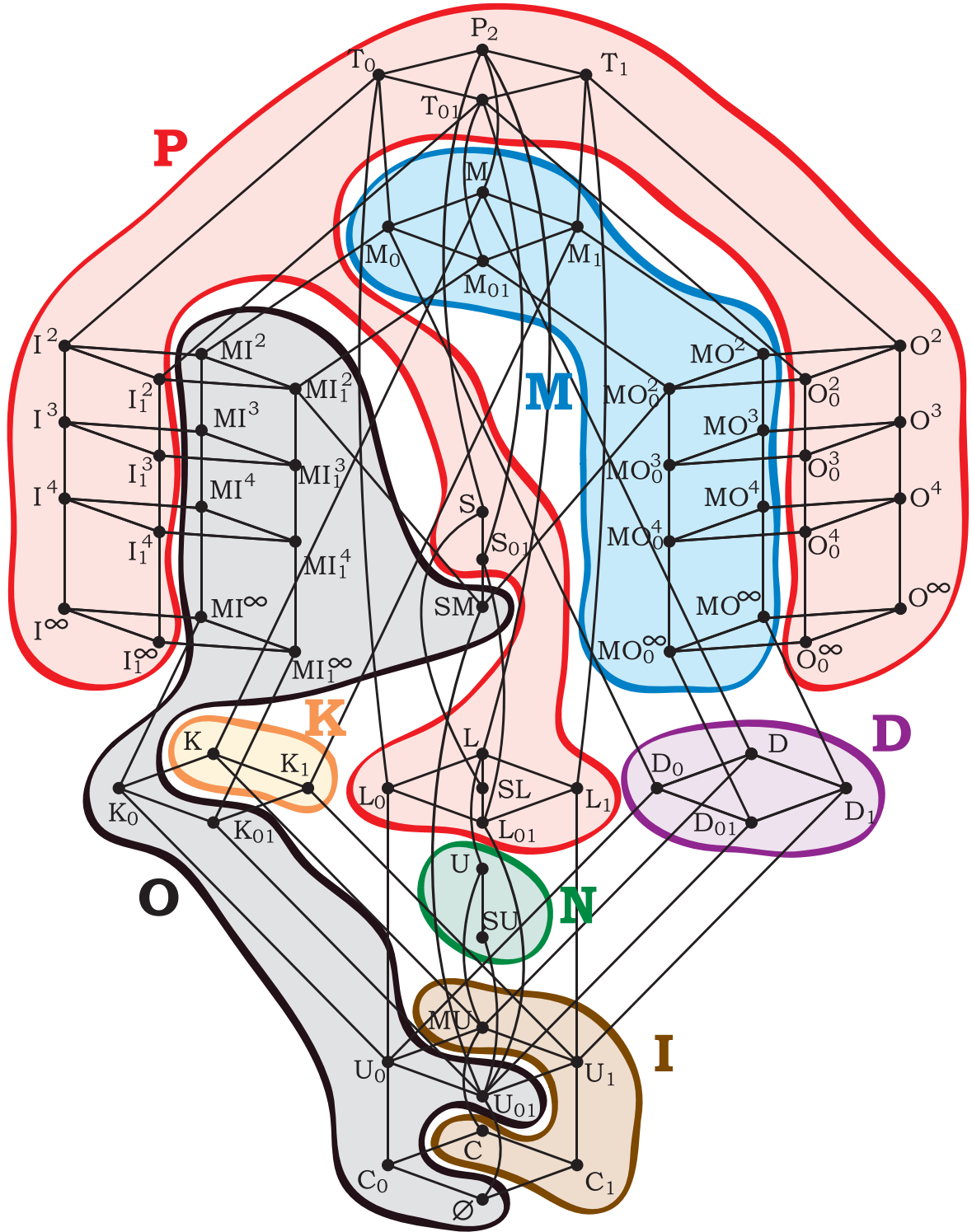
Figure 1: Seven families of language equations over Post's lattice.

Four Post's classes generating this family of languages are shown in Figure 1. We shall now see that a few other classes of Boolean functions define families that are no less natural than this one.

M: Disjunction and conjunction

The next class to be considered is defined by language equations with union and intersection. The corresponding Post's class is $M_{01} = [x \vee y, x \wedge y]$; however, more classes actually generate the same language family.

Theorem 2. *Let $\mathcal{F}$ be a class of Boolean functions, for which $[x \vee (y \wedge z)] \subseteq [\mathcal{F}] \subseteq [x \vee y, x \wedge y, 0, 1]$. Then unique solutions of systems (*) with Boolean operations from $\mathcal{F}$, concatenation and singleton constants define the conjunctive languages.*

The upper bound is given by Post's class $M = [x \vee y, x \wedge y, 0, 1]$, which contains all monotone Boolean functions. The lower bound is $MO_0^\infty = [x \vee (y \wedge z)]$. Two of the eight Post's infinite hierarchies are located between these classes, and, with respect to language equations, they collapse as shown in Figure 1.

The family of languages specified by these equations is generated by *conjunctive grammars*, introduced by Okhotin [2001, 2002], which are a generalization of the context-free grammars with rules of the form $A \rightarrow \alpha_1 \& \ldots \& \alpha_m$, where each α_i is a string comprised of symbols and variables. Conjunctive grammars have a greater expressive power than the context-free grammars: they can generate many key abstract examples of non-context-free languages, such as $\{a^n b^n c^n \mid n \geqslant 0\}$, $\{ww \mid w \in \{a, b\}^*\}$, $\{(wc)^{|w|} \mid w \in \{a, b\}^*\}$ and $\{a^{2^n} \mid n \geqslant 0\}$ (see Jeż [2007] for the latter grammar), and more. On the other hand, the main context-free parsing algorithms, such as the Cocke–Kasami–Younger, the recursive descent and the generalized LR, can be extended to conjunctive grammars, maintaining the low complexity of the originals. This makes conjunctive grammars a theoretically and practically appealing generalization of context-free grammars, A survey of these grammars, their known properties and their open problems was recently given by Okhotin [2007].

In the context of language equations, let us use the symbol M to refer to this family.

P: All Boolean operations

Language equations with all Boolean operations, that is, over the basis $P_2 = [\{x \vee y, \neg x\}]$, were studied by Okhotin [2003, 2005b]. These equations may have no solutions or multiple pairwise incomparable solutions, and the problems of testing solution existence and solution uniqueness are co-RE-complete and Π_2-complete, respectively. The family of languages representable by unique solutions of these systems is exactly the family of recursive languages.

Subsequent research done by Okhotin [2006] has shown that some restricted sets of Boolean operations are sufficient to generate all recursive languages: this is, most notably, the symmetric difference alone, or, in the terminology of the present paper, Post's class of sum modulo 2. Using this result, the following theorem can be established:

Theorem 3. *Unique solutions of systems (*) with Boolean operations from a class $\mathcal{F}$, concatenation and singleton constants generate all recursive languages if and only if any of the following three functions is in $[\mathcal{F}]$: $x \vee (y \wedge \neg z)$, $x \wedge (y \vee \neg z)$ or $x \oplus y \oplus z$.*

Let us call this family P, after the name P_2 for the class of all Boolean functions (which apparently refers to two-valued propositional logic). This family captures the notion of effective computability.

N: Negation only

The family of language equations (*) using concatenation and complementation has first been studied by Leiss [1994], who has constructed an interesting example of such an equation over a unary alphabet

$$X = a \cdot \overline{\overline{\overline{X^2}^2}^2},$$

which has a nonregular unique solution $\{a^n \mid \exists k \geqslant 0 : 2^{3k} \leqslant n < 2^{3k+2}\}$.

This family has recently been systematically studied by Okhotin and Yakimova [2006]. The problem whether a system of this kind has a solution is NP-complete. Unique solutions of such equations are generated by Boolean grammars, see Okhotin [2007], and hence are contained in $\mathrm{DTIME}(n^3)$. It was shown that the regular language $a\Sigma^*b \cup b\Sigma^*a \cup \varepsilon$ cannot be represented by such equations. Even if all regular constants are allowed in equations, the linear context-free language $(a\Sigma^*b \cup b\Sigma^*a \cup \varepsilon) \setminus \{a^n b^n \mid n > 1\}$ is not representable.

Let us position this class in the Post's lattice.

Theorem 4. *Let $\mathcal{F}$ be a class of Boolean functions, such that $[\mathcal{F}] = [\neg x]$ or $[\mathcal{F}] = [1, \neg x]$. Then unique solutions of systems (*) with Boolean operations from $\mathcal{F}$, concatenation and singleton constants define the same single class of languages.*

Two Post's classes mentioned in this theorem are denoted SU and U, respectively, which refers to self-dual unary and general unary functions. This designation appears irrelevant for language equations, hence a new name has to be invented for this family. Let us call it N, which stands for "negation".

O: No Boolean operations

It remains to consider three trivial families. The family O corresponds to equations in which no Boolean operations are allowed. It is easy to see that their unique solutions contain only singleton languages and empty sets. However, it turns out that some larger Post's classes cannot generate any more languages.

Theorem 5. *Let $\mathcal{F}$ be a class of Boolean functions, such that $[\mathcal{F}] \subseteq [(x \vee y) \wedge (y \vee z) \wedge (x \vee z)]$. Then unique solutions of systems (*) with Boolean operations from $\mathcal{F}$, concatenation and singleton constants contain languages of the form $\varnothing$ and $\{w\}$.*

Post's class $[\{(x \vee y) \wedge (y \vee z) \wedge (x \vee z)\}]$ is called MI^2. Note that conjunction is in MI^2, while constant 1 is not there. Let us denote this bottom family by O.

I: Constant 1 only

Post's class formed by constant 1 is denoted C_1, and this constant can be used to express Σ^*, which results in a somewhat larger family of languages than O. This family has the following characterization:

Theorem 6. *Let $\mathcal{F}$ be a class of Boolean functions, such that $1 \in [\mathcal{F}] \subseteq [0, 1, x]$. Then unique solutions of systems (*) with Boolean operations from $\mathcal{F}$, concatenation and singleton constants contain languages of the form $\varnothing$ and $w_0\Sigma^*w_1\Sigma^* \ldots w_{n-1}\Sigma^*w_n$, where $n \geqslant 0$ and $w_i \in \Sigma^*$.*

Let us denote this family by I; this symbol is meant to resemble the digit "1". Note that I is not closed under intersection, since the language $\Sigma^*a\Sigma^* \cap \Sigma^*b\Sigma^*$ cannot be represented in the above form.

K: Conjunction and constant 1

In the definition of O we have seen that conjunction alone does not help to generate anything more than singletons. However, once Σ^* can be expressed, the intersection operation gives a little increase to the expressive power:

Theorem 7. *Let $\mathcal{F}$ be a class of Boolean functions, such that $[\mathcal{F}] = [x \wedge y]$ or $[\mathcal{F}] = [x \wedge y, 1]$. Then unique solutions of systems (*) with Boolean operations from $\mathcal{F}$, concatenation and singleton constants contain exactly the languages from the intersection and concatenation closure of I.*

Denote this family by K. This family is distinct from I, since I is not closed under intersection.

Summary

As the reader can see in Figure 1, the above Theorems 1–7 cover the entire Post's lattice. Hence, no families besides these seven families can be generated by language equations of the given kind.

The following table attempts to review the work on these families. The column "discovered by" records the first mention of the language family in the literature, irrespective of the language equations generating this family. The next column "equations studied by" refers to the study of language equations with the corresponding operations.

	language family	discovered by	equations studied by
P	recursive	Turing (1936), Church (1936)	Okhotin [2003, 2006]
M	conjunctive	Okhotin [2001], also Szabari [1991]	Okhotin [2002]
D	context-free	Chomsky (1956), also Panini (ca. 4th cent. B. C.)	Ginsburg and Rice [1962]
N	*(no name)*	Okhotin and Yakimova [2006], also Leiss [1994]	Okhotin and Yakimova [2006]
K	$(\cdot, \cap)$-closure of I	–	Okhotin [2005a]
I	$\varnothing$, $w_0\Sigma^*w_1 \ldots \Sigma^*w_n$	–	–
O	$\varnothing$, $\{w\}$	–	–

The family P (the recursive languages) corresponds to the notion of effective computability, which was developed by Church and Turing.

The family M (the conjunctive languages) was first studied by Okhotin [2001]. An equivalent definition of such grammars was given in an unpublished Master's thesis by Szabari

[1991], though no properties of these grammars were obtained. For a survey of these grammars, see Okhotin [2007]. The latest contribution to the study of this family was made by Jeż [2007].

The family D (the context-free languages) was known since the ancient times, when the Indian legendary scholar Panini used similar formalisms to write down the grammar of Sanskrit. This family was first mathematically considered by Chomsky, and the representation by equations was discovered by Ginsburg and Rice [1962].

The family N was proposed by Leiss [1994], who constructed an interesting example and established some results for a restricted case. The first systematic study is due to Okhotin and Yakimova [2006]. The only known definition of this family is by language equations.

The following proper inclusions between these families can be established:

$$O \subset I \subset K \subset D \subset M \subset P$$
$$I \subset N \subset P$$

In addition, it is known from Okhotin and Yakimova [2006] that N is distinct from K, D and M, hence all seven families are pairwise distinct.

Conclusion

This work began a classification of language equations with concatenation and Boolean operations. The proposed method of study based upon Post's lattice is certainly applicable to larger classes of equations, such as the *unresolved equations* of the form $\varphi_i(X_1, \ldots X_n) = \psi_i(X_1, \ldots X_n)$. It remains to carry out these extended classifications. Other interesting kinds of language equations, such as equations with non-standard word operations, await new methods of classification.

Let us conclude by saying that hierarchies of language equations, such as the one established in this paper, constitute an alternative to Chomsky's hierarchy of transformational grammars. While the context-free languages can be naturally represented in both hierarchies, this is not so for many other families, and the proposed hierarchy leads to noteworthy families that would be hard to fit in the Chomskian paradigm. Furthermore, some natural families of language equations that did not fit into the present preliminary classification represent other important language families, such as the regular languages first characterized by Bondarchuk [1963], and the recursively enumerable languages and their complements, see Kunc [2005] and Okhotin [2005a, 2005b]. More research on elaborating these hierarchies will bring further interesting results in formal language theory and its applications.

References

[1963] V. G. Bondarchuk, "Sistemy uravnenii v algebre sobytii" (Systems of equations in the event algebra), in Russian, *Zhurnal vychislitel'noi matematiki i matematicheskoi fiziki* (Journal of Computational Mathematics and Mathematical Physics), 3:6, 1963.

[1962] S. Ginsburg, H. G. Rice, "Two families of languages related to ALGOL", *Journal of the ACM*, 9 (1962), 350–371.

[2007] A. Jeż, "Conjunctive grammars can generate non-regular unary languages", *Developments in Language Theory* (DLT 2007, Turku, Finland, July 3–6, 2007), to appear.

[1994] L. Kari, "On language equations with invertible operations", *Theoretical Computer Science*, 132 (1994), 129–150.

[2005] M. Kunc, "The power of commuting with finite sets of words", *STACS 2005* (Stuttgart, Germany), LNCS 3404, 569–580.

[2007] M. Kunc, "What do we know about language equations?", *Developments in Language Theory* (DLT 2007, Turku, Finland, July 3–6, 2007), to appear.

[1994] E. L. Leiss, "Unrestricted complementation in language equations over a one-letter alphabet", *Theoretical Computer Science*, 132 (1994), 71–93.

[2001] A. Okhotin, "Conjunctive grammars", *Journal of Automata, Languages and Combinatorics*, 6:4 (2001), 519–535.

[2002] A. Okhotin, "Conjunctive grammars and systems of language equations", *Programming and Computer Software*, 28 (2002), 243–249.

[2003] A. Okhotin, "Decision problems for language equations with Boolean operations", *Automata, Languages and Programming* (ICALP 2003, Eindhoven, The Netherlands, June 30–July 4, 2003), LNCS 2719, 239–251.

[2005a] A. Okhotin, "Unresolved systems of language equations: expressive power and decision problems", *Theoretical Computer Science*, 349:3 (2005), 283–308.

[2005b] A. Okhotin, "Strict language inequalities and their decision problems", *Mathematical Foundations of Computer Science* (MFCS 2005, Gdańsk, Poland, August 29–September 2, 2005), LNCS 3618, 708–719.

[2006] A. Okhotin, "Language equations with symmetric difference", *Computer Science in Russia* (CSR 2006, St. Petersburg, Russia, June 8–12, 2006), LNCS 3967, 292–303.

[2006] A. Okhotin, O. Yakimova, "On language equations with complementation", *Developments in Language Theory* (DLT 2006, Santa Barbara, USA, June 26–29, 2006), LNCS 4036, 420–432.

[2007] A. Okhotin, "Nine open problems for conjunctive and Boolean grammars", *Bulletin of the EATCS*, 91 (2007), 96–119.

[1941] E. L. Post, *The Two-Valued Iterative Systems of Mathematical Logic*, Princeton University Press, 1941.

[1991] A. Szabari, *Alternujúce Zásobníkové Automaty* (Alternating Pushdown Automata), in Slovak, diploma work (M.Sc. thesis), University of Košice (Czechoslovakia), 1991, 45 pp.

[1966] S. V. Yablonski, G. P. Gavrilov, V. B. Kudryavtsev, *Funktsii algebry logiki i klassy Posta* (Functions of the logic algebra and the classes of Post), Nauka, Moscow, 1966, in Russian.

- German translation: *Boolesche Funktionen und Postsche Klassen*, Akademie-Verlag, Berlin, 1970.