Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

Peter Kostolányi

Autoreferát dizertačnej práce

Balanced Use of Resources in Computations

na získanie akademického titulu philosophiæ doctor
v odbore doktorandského štúdia: 9.2.1. informatika

Bratislava, 10. marca 2017

Dizertačná práca bola vypracovaná v dennej forme doktorandského štúdia na Katedre informatiky FMFI UK.

Predkladateľ: RNDr. Peter Kostolányi
Katedra informatiky
Fakulta matematiky, fyziky a informatiky UK
Mlynská dolina
842 48 Bratislava

Školiteľ: prof. RNDr. Branislav Rovan, PhD.
Katedra informatiky
Fakulta matematiky, fyziky a informatiky UK
Mlynská dolina
842 48 Bratislava

Študijný odbor: 9.2.1. informatika; študijný program: informatika.

Predseda odborovej komisie:
prof. RNDr. Rastislav Kráľovič, PhD.
Katedra informatiky
Fakulta matematiky, fyziky a informatiky UK
Mlynská dolina
842 48 Bratislava

Introduction

The presented thesis deals with balanced use of resources in the broadest possible sense. Balanced use of resources is an important aspect of computations (or more generally, of discrete systems) that is considered in a variety of practical settings. Let us for instance mention certain methods of heat reduction in microprocessor design [58, 104] or load balancing in parallel computations [126]. An effort to theoretically understand balanced use of resources as a phenomenon, without any particular application in mind, has recently been initiated in the work of Kováč [84, 85] and of the present author [74, 75, 76], in which the use of states and transitions has been studied mainly for deterministic finite automata. Automata with balanced use of resources are called *equiloaded* in [84, 85, 74, 75, 76].

We continue in the theoretical direction of [84, 85, 74, 75, 76] in the presented thesis. We consider two fundamental problems. First, we deal with the question how balanced use of resources can be defined and (up to some level) studied independently of a particular model or setting. We provide a possible answer to this question in Chapter 1, in which we develop an abstract framework for studying equiloadedness based on transition systems. Second, it has been our aim to "lift" the theory for finite automata, developed in [84, 85, 74, 75, 76], to some more powerful models. In particular, we deal with balanced use of resources in certain families of context-free grammars in the remaining chapters. In order to prove our main results, we make use of methods of analytic combinatorics [43], especially of singularity analysis [42], the so-called Drmota-Lalley-Woods theorem [43, 25, 88, 125], and the recent Banderier-Drmota theorem [5] (we coin this term, as to our best knowledge we are the first to make significant use of this result).

Structure of the Thesis

- In *Chapter 1*, we develop an abstract framework for studying balanced use of resources at an abstract level of transition systems. We prove some basic results at this abstract level as well.
- In *Chapter 2*, we provide the necessary background in the theory of context-free grammars, algebraic systems, and analytic combinatorics. The material presented here is organised in a way, which reflects our approach to the analysis of combinatorial aspects of context-free grammars that we adopt later on. We give a detailed treatment of the recent Banderier-Drmota theorem [5] in this chapter, which is a cornerstone for our later developments. Our presentation also attempts to remedy several issues that seem to occur in the Banderier's and Drmota's article [5].
- In *Chapter 3*, we make some preliminary observations that we use later on to develop our theory of equiloaded context-free grammars.
- In *Chapter 4*, we study equiloaded *linear* and *ultralinear* grammars.
- In *Chapter 5*, we "lift" the theory developed in Chapter 4 to general context-free grammars.

The dependence diagram of particular chapters is shown in Figure 1.

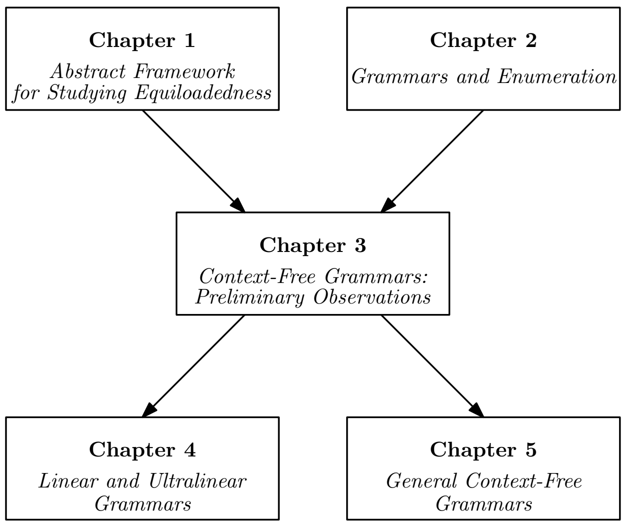

Figure 1: Dependencies between particular chapters.

1 The Abstract Framework Based on Transition Systems

In Chapter 1, we develop an abstract framework for studying balanced use of resources, which makes it possible to come up with definitions and basic results independent of a particular model or setting. This allows us to state our basic definitions once for all, without a need to do the same work repeatedly for each particular model. The "universal" nature of the definitions moreover guarantees consistency when switching between the same concepts for different models.

Previous research on theoretical aspects of balanced use of resources has mostly dealt with some simple families of automata or machines [84, 85, 74, 75, 76]. In the later chapters of the presented thesis, we mostly deal with their grammatical (or let us say equational) counterparts. We would also like to keep a possibility for our theory to be shifted to more realistic models of computers, programs, and so on. We therefore find ourselves in search of an abstraction that comprises all these particular settings.

Some sort of configuration space is present in each of the particular settings described above. Switching between configurations can be described in discrete time steps and is controlled by some relatively simple regulatory mechanism. For instance, one may think of Turing machines and their configurations "controlled" by the transition function, or of grammars and their sentential forms "controlled" by the production rules. An abstraction of this property can be provided by the concept of *transition systems* [71] (see Definition 1.1.1 in the thesis), which we use as a basis for our abstract theory of equiloadedness. Transition systems have been used in a variety of modifications, mostly in formal verification and model checking [4]. However, our use and interpretation of these structures is different in several aspects.

In Section 1.2, we show that common models of automata, machines, and grammars can all be described in terms of transition systems. Moreover, it should be at least intuitively clear that practically any relevant discrete time setting can be described in this framework. As a result, it is sufficient to provide the fundamental definitions of equiloadedness for transition systems. Definitions of equiloadedness for particular models, such as automata or grammars, can simply be obtained by specialisation of the abstract definition to the corresponding class of transition systems.

Given the framework of transition systems, in Section 1.3 we turn our attention to the problem of capturing the notion of resources. When it comes to studying equiloadedness, we are mostly interested in resources like nonterminals or production rules in context-free grammars. Our understanding of "resource" should be general enough to capture these. It should also be compatible with the resources often studied for machine-like models, such as time or space [98]. However, the concept of a resource seems too elusive to be itself formally defined. For this reason, we only formalise valid means of quantifying resources, which we call *complexity measures*. This term is inspired by complexity measures such as time or space. Nevertheless, let us emphasise once again that our main focus is on complexity measures that are slightly less usual.

We define complexity measures to be functions that assign an element of some set V to each run in a given transition system. For the most important complexity measures, one can typically take $V = \mathbb{N} \cup \{\infty\}$. This concept is subsequently extended to complexity measures on classes of transition systems. For more details, see Definition 1.3.1 and Definition 1.3.3 in the thesis.

However, complexity measures themselves do not appear to be sufficient in order to develop a full-fledged abstract theory of equiloadedness, mainly because they can only be used to assign complexities to *runs* in the transition system. For this reason, we introduce the concept of *run assignment schemata* in Section 1.4. Our definition of this concept is motivated by several aspects of complexity measures.

First motivation behind this concept originates in the theory of computational complexity. There, complexity of a computation can be used to study complexity of an input word. For deterministic models of computation, the complexity of a word is simply defined to be the complexity of the unique computation on that word. For nondeterministic models of computation, where more than one computation may exist on a single input, some kind of operation (usually a minimum) is used to determine the complexity of the input word based on complexities of corresponding computations.

Another aspect that motivated our definition of run assignment schemata is that one is often not interested solely in complexity of a single run or a single input word, but in complexity of some set of runs or words. In computational complexity theory, one usually assigns to each n the maximum among complexities of words of length up to n, and then studies the behaviour of these values when n tends to infinity. Similarly, we are often interested in complexities of sets of runs having the same length.

Finally, one is usually interested in some restricted set of runs in a transition system only. For instance, if a transition system corresponds to a finite automaton, one may be interested in runs beginning in an initial state, in runs ending in a final state, and so on. Run assignment schemata can also be used to specify the set of runs one is interested in.

We have defined a *simple run assignment schema* to be a structure containing some set of instances J, a correspondence mapping f, and an evaluator function Ψ (see Definition 1.4.1 for formal details). Instances should be thought of as objects, complexities of which one wants to study. The correspondence mapping

f relates each instance x to a set of runs $f(x)$ corresponding to x. Finally, the evaluator function Ψ takes a multiset of complexities on the input and returns one single complexity on the output. The idea is that each "compatible" complexity measure ϕ can be extended to the run assignment schema so that the complexity of each instance x in J is obtained by first applying the correspondence mapping f to x, and then by applying Ψ on the multiset of complexities of runs in $f(x)$. This is made precise in Definition 1.4.4. Moreover, we define general *run assignment schemata* and extensions of complexity measures to such schemata in a similar manner, but we allow more than one level of correspondence. That is, instances from some set J_1 may correspond to instances from some set J_2, and so on. See Definition 1.4.7 and Definition 1.4.9 in the thesis.

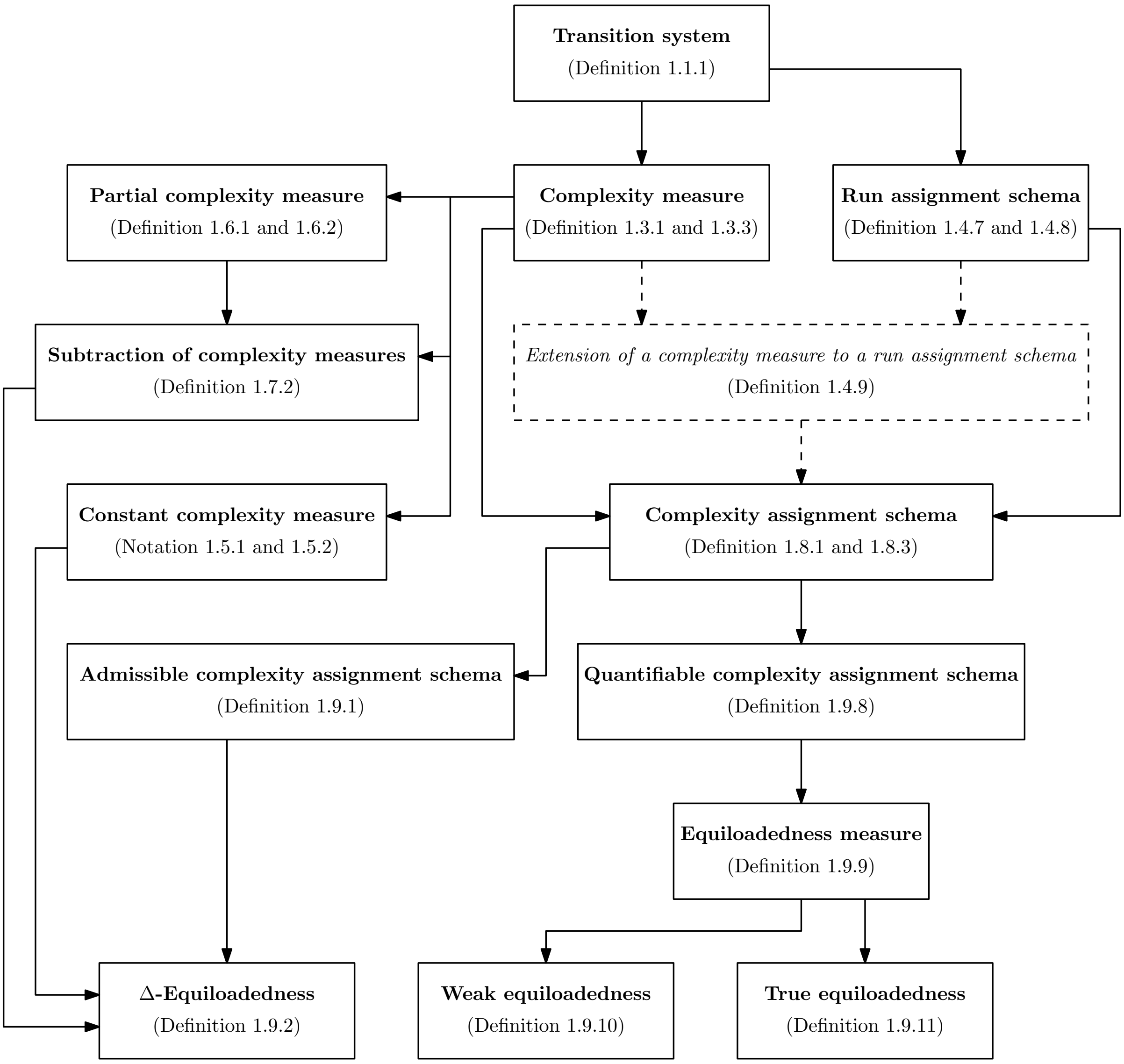

Figure 2: The genesis of our abstract definitions of equiloadedness.

From Section 1.5 to Section 1.7, we develop a technical toolkit needed to formulate our definitions of equiloadedness. For this reason, we introduce *constant* and *bounded* complexity measures, *partial complexity measures*, and some operations on complexity measures.

In Section 1.8, we introduce *complexity assignment schemata* (see Definition 1.8.1). A complexity assignment schema is a run assignment schema together with a set of complexity measures. All definitions of equiloadedness that we propose in the thesis are formulated relative to a complexity assignment schema. In essence, the idea always is to take all complexity measures from the complexity assignment schema and extend them to the given run assignment schema. If the resulting complexities are close enough one to each other (the precise meaning of this depends on a particular definition of equiloadedness), then the use of resources quantified by the complexity measures in consideration can be said to be balanced.

In Section 1.9, we finally define equiloaded transition systems over a given complexity assignment schema. In particular, we introduce three types of equiloadedness: Δ-*equiloadedness* (Definition 1.9.2), *weak equiloadedness* (Definition 1.9.10), and *true equiloadedness* (Definition 1.9.11). These can be viewed as generalisations of the notions of equiloadedness used in [84, 85, 74, 75, 76] to study balanced use of resources in finite automata (see the notes at the end of Chapter 1 for details). There are some technical assumptions needed in our definitions of equiloadedness. In particular, we define Δ-equiloadedness just for complexity assignment schemata that we call *admissible* (see Definition 1.9.1), and we define weak equiloadedness and true equiloadedness just for complexity assignment schemata that we call *quantifiable* (see Definition 1.9.8).

The interconnections between the notions that we have used to formulate our abstract definitions of equiloadedness are depicted graphically in Figure 2.

In the rest of the first chapter, we focus on some constructions on run assignment schemata and complexity assignment schemata, which keep certain equiloadedness properties unchanged. More precisely, we deal with what we call *restrictions*, *refinements*, and *additive combinations*.

2 Context-Free Grammars and Enumeration

In Chapter 2, we give an overview of concepts and known results that seem necessary to be stated and provide the reader with an idea about some key points of the methodology that we adopt in the remaining chapters. We do so mainly through specific organisation of the material, which is designed to emphasise the close relations between grammars and some standard methods of combinatorial enumeration.

In Section 2.1, we identify the place occupied by automata and grammars in the bigger picture of analysis of systems. We demonstrate that (at least deterministic) automata can essentially be seen as dynamical systems. On the other hand, grammars can be seen as equations – fixed point equations, if the grammar is context-free. The prominent role played by various kinds of automata and grammars in formal language theory [56, 66] can thus be linked to the prominent role of dynamics and fixed points in applied mathematics as a whole [70, 54].

Later in Chapter 2, we survey the basics of the theory of formal power series (noncommutative, commutative, and over graded monoids), list some basic facts related to context-free grammars and algebraic systems, and finally introduce some basic methods of analytic combinatorics [43] and especially of singularity analysis of Flajolet and Odlyzko [42]. In particular, we explore in Section 2.8 the recent result of Banderier and Drmota [5] characterising Puiseux expansions of $\mathbb{N}$-algebraic functions. Our presentation of this result slightly differs from the original presentation of Banderier and Drmota [5], mainly due to our effort to fix several issues that seem to occur in [5].

3 Preliminary Observations for Context-Free Grammars

The principal aim of the presented thesis is to study various families of equiloaded context-free grammars. In Chapter 3, we introduce several concepts and prove a number of auxiliary results that we utilise later in Chapter 4 and in Chapter 5 to obtain our main results.

In Section 3.1, we describe a canonical construction of a transition system that captures a given context-free grammar and its leftmost derivations. Moreover, we introduce several complexity assignment schemata on such transition systems. Most importantly, we give definitions of schemata $(\mathfrak{FoL}, \mathfrak{Prod})$, $(\mathfrak{FoL}, \mathfrak{NProd})$, and $(\mathfrak{FoL}, \mathfrak{Non})$, which provide suitable frameworks for counting applications of particular production rules, non-terminating production rules, and nonterminals in full[1] leftmost derivations of a given length (see Notation 3.1.4 and Notation 3.1.6). In the rest of the thesis, we use the constructions described in Section 3.1 *implicitly*. This means that we deal directly with context-free grammars, but we bear in mind the fact that we are in fact dealing with some special families of transition systems. This point of view makes it possible to transfer the notions defined for transition systems in Chapter 1, such as our definitions of equiloadedness, to context-free grammars. As a result, the setting of context-free grammars can be viewed as a particular specialisation of the abstract setting described in Chapter 1.

In order to study equiloadedness of context-free grammars over schemata $(\mathfrak{FoL}, \mathfrak{Prod})$, $(\mathfrak{FoL}, \mathfrak{NProd})$, and $(\mathfrak{FoL}, \mathfrak{Non})$, we develop some methods for combinatorial analysis of quantities such as the number of full leftmost derivations of length n in a given grammar $\mathcal{G}$ or the number of applications of a given production rule in full leftmost derivations of length n in $\mathcal{G}$. The approach that we follow when dealing with problems like these is to transform the grammar $\mathcal{G}$ so that the resulting grammar is unambiguous and the number of words of length n generated by this grammar is equal to the quantity in consideration. After such transformation, methods based on the Banderier-Drmota theorem can be applied to obtain an asymptotic estimate. In order to count the number of full leftmost derivations, it is possible to utilise the already well known concept of *left Szilard languages* [91, 92, 93]. The number of words of length n in the left Szilard language for some grammar $\mathcal{G}$ is always equal to the number of full leftmost derivations of length n in $\mathcal{G}$. We call the canonical grammar

[1] By a *full leftmost derivation*, we mean a leftmost derivation beginning in the initial nonterminal and ending with a terminal word.

generating the left Szilard language the *left Szilard grammar* (these grammars are described in Section 3.2). To count applications of production rules in full leftmost derivations, we introduce in Section 3.3 what we call *pebble distribution grammars* (see Definition 3.3.1). The number of words of length n generated by the pebble distribution grammar for a grammar $\mathcal{G}$ and its production rule π equals the number of applications of π in full leftmost derivations of length n in $\mathcal{G}$. Moreover, pebble distribution grammars are always unambiguous. We also relate pebble distribution grammars to Jacobians of algebraic systems (Theorem 3.3.4).

In Section 3.4, we define the notion of an *enriched dependence hypergraph* for a given context-free grammar $\mathcal{G}$. This is a central concept for our considerations, as our main results are characterisations of families of equiloaded grammars via some properties of their enriched dependence hypergraphs.

From Section 3.5 to Section 3.7, we focus on the case in which the generating functions for our quantities are periodic (that is, they have multiple dominating singularities). The possibility of periodic behaviour makes our considerations slightly more complicated, as the Banderier-Drmota theorem can be *directly* applied to obtain an asymptotic estimate only if the function is aperiodic. For this reason, we provide a framework that we later apply to handle the periodic case as well.

In Section 3.8, we make some connections to Perron-Frobenius theory. Finally, in Section 3.9, we link certain properties of generating functions counting production rule applications in full leftmost derivations to the properties of generating functions counting full leftmost derivations themselves. We apply these results later in Chapter 4 and Chapter 5.

4 Results for Ultralinear Grammars

In Chapter 4, we study equiloaded *ultralinear* grammars. Note that each linear grammar is ultralinear and each right-linear grammar is linear. That is, the results that we prove in Chapter 4 apply to these families of grammars as well.

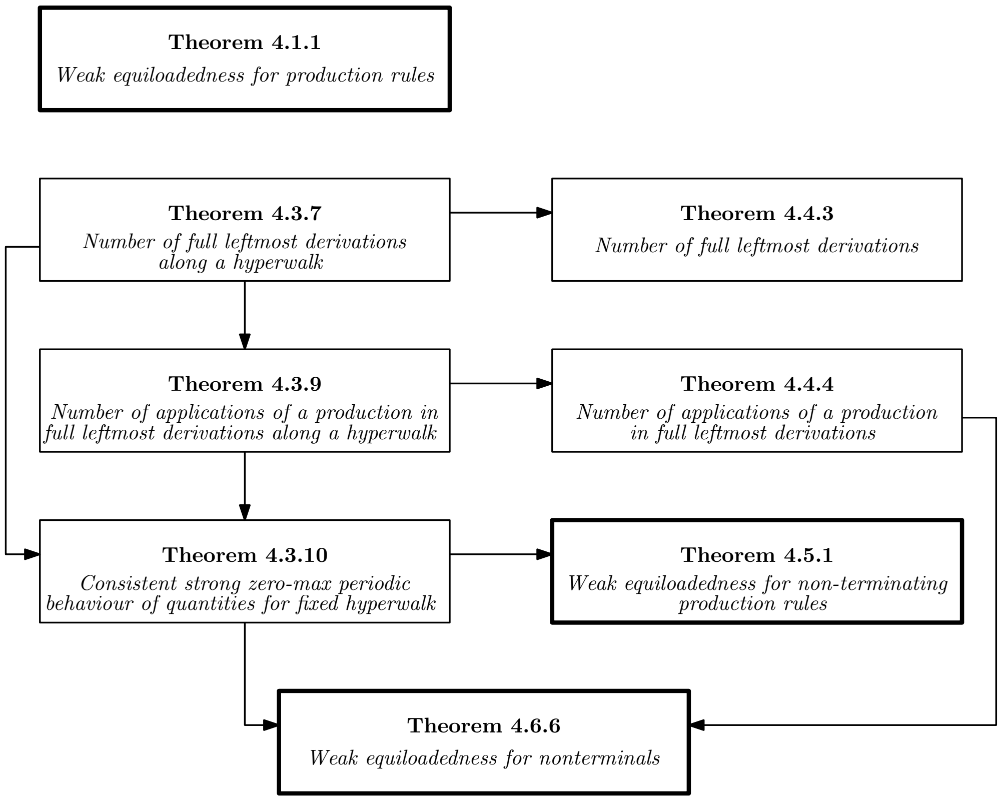

Figure 3: Interconnections between the key results of Chapter 4. Characterisations of families of equiloaded grammars are drawn in “fat” rectangles and the most important ingredients for their proofs are drawn in “ordinary” rectangles.

We show in Section 4.1 that weak equiloadedness for production rules is almost impossible for ultralinear grammars. More precisely, an ultralinear grammar is weakly equiloaded for production rules if and only if its left Szilard language is finite. Moreover, in Section 4.2 we define what we call *transient nonterminals* and prove that the situation with weak equiloadedness for nonterminals is similar to the situation with weak equiloadedness for production rules if an ultralinear grammar contains such nonterminals. In the rest of Chapter 4, we usually work under the assumption that the grammar in consideration is strongly proper (see Definition 2.3.15). The results of Section 4.2 imply that we lose nothing interesting by this assumption.

In order to characterise ultralinear grammars that are weakly equiloaded for non-terminating production rules and especially for nonterminals, some preparation is needed, which we undergo in Section 4.3 and Section 4.4. In Section 4.3, we define full leftmost derivations along a given full hyperwalk in the enriched dependence hypergraph (Definition 4.3.2 and Definition 4.3.3; see also Definition 3.4.11, which introduces the notion of a full hyperwalk). Next, we characterise leading singular expansions[2] of generating functions counting full leftmost derivations of given length along a fixed full hyperwalk (Theorem 4.3.7) and of generating functions counting applications of a given production rule in such full leftmost derivations (Theorem 4.3.9). We also prove that under some assumptions, these generating functions exhibit what we call consistent strong zero-max periodic behaviour (Theorem 4.3.10; see Section 3.6 for our definitions related to zero-max periodic behaviour). In Section 4.4, we use the results established in Section 4.3 to characterise leading singular expansions of generating functions counting all full leftmost derivations of a given length (Theorem 4.4.3) and applications of a given production rule in all full leftmost derivations of a given length (Theorem 4.4.4).

In Section 4.5, we finally characterise ultralinear grammars that are weakly equiloaded for non-terminating production rules and in Section 4.6, we prove a characterisation of ultralinear grammars weakly equiloaded for nonterminals. The interconnections between the key results of Chapter 4 are depicted in Figure 3.

Finally, we prove in Section 4.7 that for non-terminating production rules and nonterminals, true equiloadedness essentially coincides with Δ-equiloadedness (this observation is trivial for arbitrary production rules).

5 Results for General Context-Free Grammars

In Chapter 5, we essentially "lift" the theory developed in Chapter 4 for ultralinear grammars to general context-free grammars. The structure of both chapters is similar. Nevertheless, heavier machinery is needed in Chapter 5, as the generating functions corresponding to arbitrary context-free grammars are not necessarily rational, but they are ℕ-algebraic. For this reason, the Banderier-Drmota theorem and similar tools are crucial for the purposes of Chapter 5.

Throughout Chapter 5, we work with strongly proper context-free grammars only (see Definition 2.3.15). However, we argue in the introduction to the chapter that we do not lose much with this assumption.

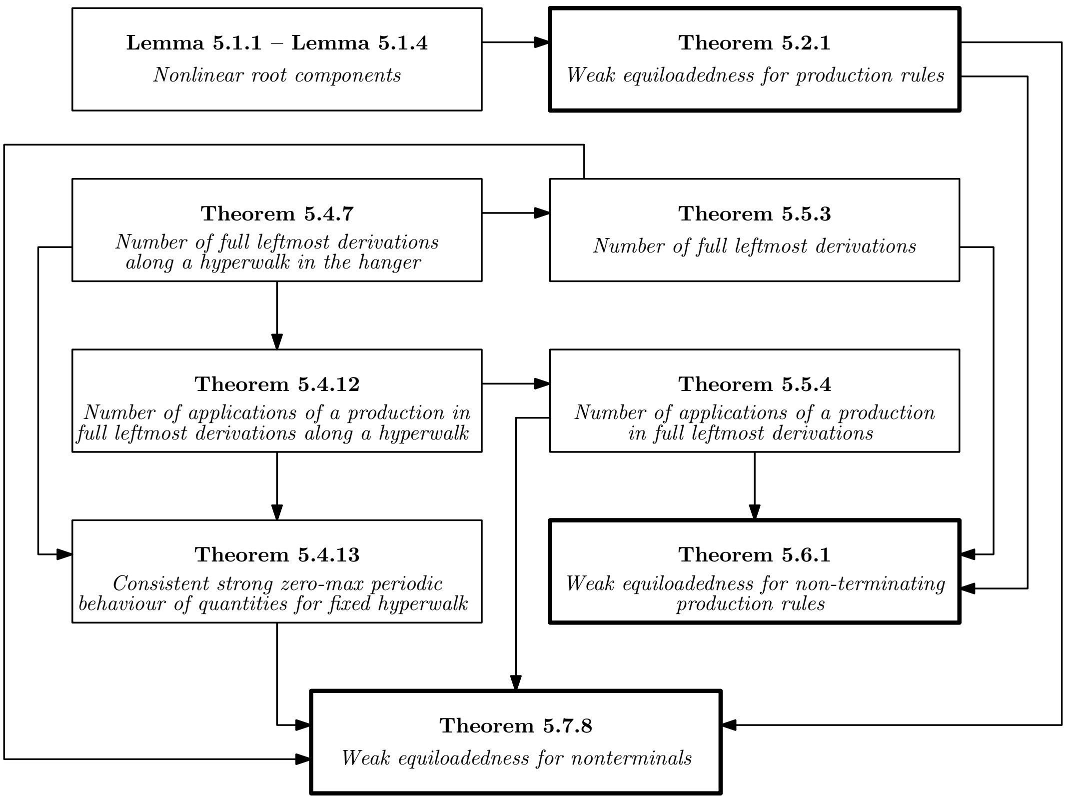

Figure 4: Interconnections between the key results of Chapter 5. Characterisations of families of equiloaded grammars are drawn in "fat" rectangles and the most important ingredients for their proofs are drawn in "ordinary" rectangles.

[2]These are Laurent expansions, as generating functions corresponding to ultralinear grammars are always rational.

In Section 5.1, we prove several auxiliary lemmas for context-free grammars with nonlinear[3] root components, which we use in Section 5.2 to prove a characterisation of context-free grammars weakly equiloaded for production rules. Similarly to its counterpart for ultralinear grammars, this is the easiest of the characterisations proved in Chapter 5. However, unlike in the ultralinear case, it turns out that the family of context-free grammars weakly equiloaded for production rules is quite rich and by no means trivial.

In Section 5.3, we introduce *hangers* (Definition 5.3.2) and *semihangers* (Definition 5.3.7) of enriched dependence hypergraphs. These are certain restrictions of enriched dependence hypergraphs that play an important role in our later considerations. Hangers appear to be fundamental objects in many respects, and we use semihangers in Section 5.7 in our characterisation of context-free grammars weakly equiloaded for nonterminals. We also introduce the notion of *vertex exponents* in Section 5.3.

In Section 5.4, we define full leftmost derivations along a given full hyperwalk in the hanger of an enriched dependence hypergraph (Definition 5.4.2 and Definition 5.4.3). This forms a generalisation of the similar concept introduced for ultralinear grammars in Section 4.3. The results proved in Section 5.4 – that is, Theorem 5.4.7, Theorem 5.4.12, and Theorem 5.4.13 – are counterparts of Theorem 4.3.7, Theorem 4.3.9, and Theorem 4.3.10 for ultralinear grammars. Similarly, the results proved in Section 5.5 – namely Theorem 5.5.3 and Theorem 5.5.4 – are counterparts of Theorem 4.4.3 and Theorem 4.4.4 for ultralinear grammars.

Finally, in Section 5.6, we use the machinery developed so far to prove a characterisation of context-free grammars weakly equiloaded for non-terminating production rules and in Section 5.7 we prove a characterisation of context-free grammars weakly equiloaded for nonterminals.

[3]This is in the terminology of algebraic systems – in the strict terminology of grammars, the term “expansive” would perhaps be more appropriate [6, 9, 23, 86, 87, 103].

Publications

KOSTOLÁNYI, P. 2015. A Pumping Lemma for Flip-Pushdown Languages. In *Non-Classical Models of Automata and Applications (NCMA 2015)*. Vienna : ÖCG, 2015, pp. 125–140.

KOSTOLÁNYI, P. 2015. Two Grammatical Equivalents of Flip-Pushdown Automata. In *Theory and Practice of Computer Science (SOFSEM 2015)*. Heidelberg : Springer, 2015, pp. 302–313.

KOSTOLÁNYI, P. 2016. A Pumping Lemma for Flip-Pushdown Languages. In *RAIRO – Theoretical Informatics and Applications*. 2017, vol. 50, no. 4, pp. 295–311.

KOSTOLÁNYI, P. – ROVAN, B. 2016. Automata with Auxiliary Weights. In *International Journal of Foundations of Computer Science*. 2016, vol. 27, no. 7, pp. 787–807.

Bibliography

[1] ABLOWITZ, M. J. – FOKAS, A. S. 2003. *Complex Variables: Introduction and Applications, 2nd ed.* Cambridge : Cambridge University Press, 2003. ISBN 978-0-521-53429-1.

[2] ALUR, R. et al. 2013. Regular Functions and Cost Register Automata. In *Logic in Computer Science (LICS 2013)*. Washington : IEEE Computer Society, 2013, pp. 13–22.

[3] ARNOLD, A. 1994. *Finite Transition Systems.* Paris, Hemel Hempstead : Masson, Prentice Hall, 1994. ISBN 0-13-092990-5.

[4] BAIER, C. – KATOEN, J. 2008. *Principles of Model Checking.* Cambridge : MIT Press, 2008. ISBN 978-0-262-02649-9.

[5] BANDERIER, C. – DRMOTA, M. 2015. Formulae and Asymptotics for Coefficients of Algebraic Functions. In *Combinatorics, Probability and Computing.* 2015, vol. 24, no. 1, pp. 1–53.

[6] BARON, G. – KUICH, W. 1981. The Characterization of Nonexpansive Grammars by Rational Power Series. In *Information and Control.* 1981, vol. 48, no. 2, pp. 109–118.

[7] BENDER, E. A. 1974. Partitions of Multisets. In *Discrete Mathematics.* 1974, vol. 9, no. 4, pp. 301–311.

[8] BERMAN, A. – PLEMMONS, R. J. 1994. *Nonnegative Matrices in the Mathematical Sciences.* Philadelphia : SIAM, 1994. ISBN 0-89871-321-8.

[9] BERSTEL, J. 1979. *Transductions and Context-Free Languages.* Stuttgart : B. G. Teubner, 1979. ISBN 3-519-02340-7.

[10] BERSTEL, J. – REUTENAUER, C. 2011. *Noncommutative Rational Series with Applications.* Cambridge : Cambridge University Press, 2011. ISBN 978-0-521-19022-0.

[11] BHATIA, N. P. – HAJEK, O. 1969. *Local Semi-Dynamical Systems.* Heidelberg : Springer, 1969. ISBN 978-3-540-04609-7.

[12] BOUSQUET-MÉLOU, M. 2007. Rational and algebraic series in combinatorial enumeration. In *Proceedings of the International Congress of Mathematicians (ICM 2006), vol. III.* Zürich : European Mathematical Society, 2007, pp. 789–826.

[13] BRIDSON, M. R. – GILMAN, R. H. 2002. Context-Free Languages of Sub-Exponential Growth. In *Journal of Computer and System Sciences.* 2002, vol. 64, no. 2, pp. 308–310.

[14] CADILHAC, M. – KREBS, A. – LIMAYE, N. 2015. Value Automata with Filters. In *Non-Classical Models of Automata and Applications (NCMA 2015). Short Papers.* Vienna : ÖCG, 2015, pp. 13–21.

[15] CHATTERJEE, K. – DOYEN, L. – HENZINGER, T. A. 2009. Alternating Weighted Automata. In *Fundamentals of Computation Theory (FCT 2009).* Heidelberg : Springer, 2009, pp. 3–13.

[16] CHATTERJEE, K. – DOYEN, L. – HENZINGER, T. A. 2009. Expressiveness and Closure Properties for Quantitative Languages. In *Logic in Computer Science (LICS 2009).* Washington : IEEE Computer Society, 2009, pp. 199–208.

[17] CHATTERJEE, K. – DOYEN, L. – HENZINGER, T. A. 2010. Quantitative Languages. In *ACM Transactions on Computational Logic.* 2010, vol. 11, no. 4, article 23.

[18] CHOMSKY, N. 1956. Three Models for the Description of Language. In *IRE Transactions on Information Theory.* 1956, vol. 2, no. 3, pp. 113–124.

[19] CHOMSKY, N. 1962. Formal Properties of Grammars. In *Handbook of Mathematical Psychology*, vol. 2. New York : John Wiley & Sons, 1962, pp. 323–418.

[20] CHOMSKY, N. – SCHÜTZENBERGER, M. P. 1963. The Algebraic Theory of Context-Free Languages. In *Computer Programming and Formal Systems.* Amsterdam : North-Holland, 1963, pp. 118–161.

[21] CORMEN, T. H. – LEISERSON, C. E. – RIVEST, R. L. – STEIN, C. 2009. *Introduction to Algorithms, 3rd ed.* Cambridge : MIT Press, 2009. ISBN 978-0-262-03384-8.

[22] COX, D. A. – LITTLE, J. – O'SHEA, D. 2015. *Ideals, Varieties, and Algorithms, 4th ed.* Cham : Springer, 2015. ISBN 978-3-319-16720-6.

[23] CREMERS, A. – GINSBURG, S. 1975. Context-Free Grammar Forms. In *Journal of Computer and System Sciences.* 1975, vol. 11, no. 1, pp. 86–117.

[24] DEVANEY, R. L. 1989. *An Introduction to Chaotic Dynamical Systems, 2nd ed.* Redwood City : Addison-Wesley, 1989. ISBN 0-201-13046-7.

[25] DRMOTA, M. 1997. Systems of Functional Equations. In *Random Structures & Algorithms.* 1997, vol. 10, no. 1–2, pp. 103–124.

[26] DROSTE, M. – KUICH, W. 2009. Semirings and Formal Power Series. In *Handbook of Weighted Automata.* Heidelberg : Springer, 2009, pp. 3–28.

[27] DROSTE, M. – KUICH, W. 2013. Weighted Finite Automata over Hemirings. In *Theoretical Computer Science.* 2013, vol. 485, pp. 38–48.

[28] DROSTE, M. – KUICH, W. – VOGLER, H. (eds.). 2009. *Handbook of Weighted Automata.* Heidelberg : Springer, 2009. ISBN 978-3-642-01491-8.

[29] DROSTE, M. – MEINECKE, I. 2010. Describing Average- and Longtime-Behavior by Weighted MSO Logics. In *Mathematical Foundations of Computer Science (MFCS 2010).* Heidelberg : Springer, 2010, pp. 537–548.

[30] DROSTE, M. – MEINECKE, I. 2011. Weighted Automata and Regular Expressions over Valuation Monoids. In *International Journal of Foundations of Computer Science.* 2011, vol. 22, no. 8, pp. 1829–1844.

[31] DROSTE, M. – MEINECKE, I. 2012. Weighted Automata and Weighted MSO Logics for Average and Long-Time Behaviors. In *Information and Computation.* 2012, vol. 220, pp. 44–59.

[32] DROSTE, M. – PEREVOSHCHIKOV, V. 2013. Multi-Weighted Automata and MSO Logic. In *Computer Science – Theory and Applications (CSR 2013).* Heidelberg : Springer, 2013, pp. 418–430.

[33] DROSTE, M. – PEREVOSHCHIKOV, V. 2016. Multi-Weighted Automata and MSO Logic. In *Theory of Computing Systems.* 2016, vol. 59, no. 2, pp. 231–261.

[34] DROSTE, M. – SAKAROVITCH, J. – VOGLER, H. 2008. Weighted Automata with Discounting. In *Information Processing Letters*. 2008, vol. 108, no. 1, pp. 23–28.

[35] EILENBERG, S. 1974. *Automata, Languages, and Machines, Vol. A*. New York : Academic Press, 1974. ISBN 0-12-234001-9.

[36] ELAYDI, S. 2005. *An Introduction to Difference Equations, 3rd ed.* New York : Springer, 2005. ISBN 0-387-23059-9.

[37] ÉSIK, Z. 2009. Fixed Point Theory. In *Handbook of Weighted Automata*. Heidelberg : Springer, 2009, pp. 29–65.

[38] ÉSIK, Z. – KUICH, W. 2014. On Power Series over a Graded Monoid. In *Computing with New Resources*. Cham : Springer, 2014, pp. 49–55.

[39] EVEY, R. J. 1963. Application of Pushdown-Store Machines. In *Joint Computer Conference (AFIPS '63, Fall)*. New York : ACM, 1963, pp. 215–227.

[40] FILIOT, E. – GENTILINI, R. – RASKIN, J. 2012. Quantitative Languages Defined by Functional Automata. In *Concurrency Theory (CONCUR 2012)*. Heidelberg : Springer, 2012, pp. 132–146.

[41] FLAJOLET, P. 1987. Analytic Models and Ambiguity of Context-Free Languages. In *Theoretical Computer Science*. 1987, vol. 49, pp. 283–309.

[42] FLAJOLET, P. – ODLYZKO, A. 1990. Singularity Analysis of Generating Functions. In *SIAM Journal on Discrete Mathematics*. 1990, vol. 3, no. 2, pp. 216–240.

[43] FLAJOLET, P. – SEDGEWICK, R. 2009. *Analytic Combinatorics*. Cambridge : Cambridge University Press, 2009. ISBN 978-0-521-89806-5.

[44] FLECK, A. C. 1974. An Analysis of Grammars by Their Derivation Sets. In *Information and Control*. 1974, vol. 24, no. 4, pp. 389–398.

[45] GALLO, G. et al. 1993. Directed Hypergraphs and Applications. In *Discrete Applied Mathematics*. 1993, vol. 42, no. 2–3, pp. 177–201.

[46] GINSBURG, S. 1966. *The Mathematical Theory of Context-Free Languages*. New York : McGraw-Hill, 1966. ISBN 0-07-023280-6.

[47] GINSBURG, S. 1975. *Algebraic and Automata-Theoretic Properties of Formal Languages*. Amsterdam : North-Holland, 1975. ISBN 0-7204-2506-9.

[48] GINSBURG, S. – GREIBACH, S. A. – HARRISON, M. A. 1967. Stack Automata and Compiling. In *Journal of the ACM*. 1967, vol. 14, no. 1, pp. 172–201.

[49] GINSBURG, S. – RICE, H. G. 1962. Two Families of Languages Related to ALGOL. In *Journal of the ACM*. 1962, vol. 9, no. 3, pp. 350–371.

[50] GINSBURG, S. – SPANIER, E. H. 1964. Bounded ALGOL-Like Languages. In *Transactions of the American Mathematical Society*. 1964, vol. 113, no. 2, pp. 333–368.

[51] GINSBURG, S. – SPANIER, E. H. 1966. Finite-Turn Pushdown Automata. In *SIAM Journal on Control*. 1966, vol. 4, no. 3, pp. 429–453.

[52] GIUNTI, M. – MAZZOLA, C. 2012. Dynamical Systems on Monoids: Toward a General Theory of Deterministic Systems and Motion. In *Methods, Models, Simulations and Approaches Towards a General Theory of Change (Proceedings of the Fifth National Conference of the Italian Systems Society)*. Singapore : World Scientific, 2012, pp. 173–185.

[53] GOLAN, J. S. 1999. *Semirings and their Applications*. Dordrecht : Kluwer Academic Publishers, 1999. ISBN 978-90-481-5252-0.

[54] GRANAS, A. – DUGUNDJI, J. 2003. *Fixed Point Theory*. New York : Springer, 2003. ISBN 0-387-00173-5.

[55] GREIBACH, S. A. 1965. A New Normal-Form Theorem for Context-Free Phrase Structure Grammars. In *Journal of the ACM*. 1965, vol. 12, no. 1, pp. 42–52.

[56] HARRISON, M. A. 1978. *Introduction to Formal Language Theory*. Reading : Addison-Wesley, 1978. ISBN 0-201-02955-3.

[57] HENNIE, F. C. 1965. One-Tape, Off-Line Turing Machine Computations. In *Information and Control*. 1965, vol. 8, no. 6, pp. 553–578.

[58] HEO, S. – BARR, K. – ASANOVIĆ, K. 2003. Reducing Power Density through Activity Migration. In *Proceedings of the 2003 International Symposium on Low Power Electronics and Design (ISLPED 2003)*. Washington : IEEE Computer Society, 2003, pp. 217–222.

[59] HILLE, E. 1982. *Analytic Function Theory, Vol. I, 2nd ed.* New York : Chelsea Publishing Company, 1982. ISBN 0-8284-0269-8.

[60] HILLE, E. 2002. *Analytic Function Theory, Vol. II, 2nd ed.* Providence : AMS Chelsea Publishing, 2002. ISBN 0-8218-3344-8.

[61] HOLZER, M. – KUTRIB, M. 2003. Flip-Pushdown Automata: $k+1$ Pushdown Reversals Are Better than k. In *Automata, Languages, and Programming (ICALP 2003)*. Heidelberg : Springer, 2003, pp. 490–501.

[62] HOLZER, M. – KUTRIB, M. 2003. Flip-Pushdown Automata: Nondeterminism is Better than Determinism. In *Developments in Language Theory (DLT 2003)*. Heidelberg : Springer, 2003, pp. 361–372.

[63] HOPCROFT, J. E. – ULLMAN, J. D. 1967. An Approach to a Unified Theory of Automata. In *Bell System Technical Journal*. 1967, vol. 46, no. 8, pp. 1793–1829.

[64] HOPCROFT, J. E. – ULLMAN, J. D. 1967. An Approach to a Unified Theory of Automata. In *Switching and Automata Theory (SWAT 1967)*. Washington : IEEE Computer Society, 1967, pp. 140–147.

[65] HOPCROFT, J. E. – ULLMAN, J. D. 1967. Nonerasing Stack Automata. In *Journal of Computer and System Sciences*. 1967, vol. 1, no. 2, pp. 166–186.

[66] HOPCROFT, J. E. – ULLMAN, J. D. 1979. *Introduction to Automata Theory, Languages, and Computation*. Reading : Addison-Wesley, 1979. ISBN 0-201-02988-X.

[67] HORN, R. A. – JOHNSON, C. R. 2013. *Matrix Analysis, 2nd ed.* Cambridge : Cambridge University Press, 2013. ISBN 978-0-521-83940-2.

[68] IGARASHI, Y. 1977. The Tape Complexity of Some Classes of Szilard Languages. In *SIAM Journal on Computing*. 1977, vol. 6, no. 3, pp. 460–466.

[69] INCITTI, R. 2001. The Growth Function of Context-Free Languages. In *Theoretical Computer Science*. 2001, vol. 255, pp. 601–605.

[70] KATOK, A. – HASSELBLATT, B. 1995. *Introduction to the Modern Theory of Dynamical Systems*. Cambridge : Cambridge University Press, 1995. ISBN 0-521-34187-6.

[71] KELLER, R. M. 1976. Formal Verification of Parallel Programs. In *Communications of the ACM*. 1976, vol. 19, no. 7, pp. 371–384.

[72] KNUTH, D. E. 1998. *The Art of Computer Programming, 2nd ed. Volume 3: Sorting and Searching*. Reading : Addison-Wesley, 1998. ISBN 0-201-89685-0.

[73] KORENJAK, A. J. – HOPCROFT, J. E. 1966. Simple Deterministic Languages. In *Switching and Automata Theory (SWAT 1966)*. Washington : IEEE Computer Society, 1966, pp. 36–46.

[74] KOSTOLÁNYI, P. 2011. *Rovnomerné využívanie prechodov v konečných automatoch* : Bakalárska práca. Bratislava : Fakulta matematiky, fyziky a informatiky UK v Bratislave, 2011.

[75] KOSTOLÁNYI, P. 2013. *Balanced Use of Resources in Computations* : Diplomová práca. Bratislava : Fakulta matematiky, fyziky a informatiky UK v Bratislave, 2013.

[76] KOSTOLÁNYI, P. 2013. *Balanced Use of Resources in Computations* : ŠVOČ 2013. Bratislava : Fakulta matematiky, fyziky a informatiky UK v Bratislave, 2013.

[77] KOSTOLÁNYI, P. 2015. A Pumping Lemma for Flip-Pushdown Languages. In *Non-Classical Models of Automata and Applications (NCMA 2015)*. Vienna : ÖCG, 2015, pp. 125–140.

[78] KOSTOLÁNYI, P. 2015. *Automaty s prevracateľným zásobníkom* : Rigorózna práca. Bratislava : Fakulta matematiky, fyziky a informatiky UK v Bratislave, 2015.

[79] KOSTOLÁNYI, P. 2015. *Balanced Use of Resources in Computations* : Písomná práca k dizertačnej skúške a projekt dizertačnej práce. Bratislava : Fakulta matematiky, fyziky a informatiky UK v Bratislave, 2015.

[80] KOSTOLÁNYI, P. 2015. Two Grammatical Equivalents of Flip-Pushdown Automata. In *Theory and Practice of Computer Science (SOFSEM 2015)*. Heidelberg : Springer, 2015, pp. 302–313.

[81] KOSTOLÁNYI, P. 2016. A Pumping Lemma for Flip-Pushdown Languages. In *RAIRO – Theoretical Informatics and Applications*. 2017, vol. 50, no. 4, pp. 295–311.

[82] KOSTOLÁNYI, P. 2016. *Variácie na Kleeneho vetu a automaty s váhami* : Elektronický učebný text. Bratislava : Fakulta matematiky, fyziky a informatiky UK v Bratislave, 2016.

[83] KOSTOLÁNYI, P. – ROVAN, B. 2016. Automata with Auxiliary Weights. In *International Journal of Foundations of Computer Science*. 2016, vol. 27, no. 7, pp. 787–807.

[84] KOVÁČ, I. 2008. *O využívaní stavov v konečných automatoch* : Bakalárska práca. Bratislava : Fakulta matematiky, fyziky a informatiky UK v Bratislave, 2008.

[85] KOVÁČ, I. 2010. *Equiloaded Automata* : Diplomová práca. Bratislava : Fakulta matematiky, fyziky a informatiky UK v Bratislave, 2010.

[86] KUICH, W. 1997. Semirings and Formal Power Series: Their Relevance to Formal Languages and Automata. In *Handbook of Formal Languages, Vol. 1*. Heidelberg : Springer, 1997, pp. 609–677.

[87] KUICH, W. – SALOMAA, A. 1986. *Semirings, Automata, Languages*. Heidelberg : Springer, 1986. ISBN 978-3-642-69961-0.

[88] LALLEY, S. P. 1993. Finite Range Random Walk on Free Groups and Homogeneous Trees. In *The Annals of Probability*. 1993, vol. 21, no. 4, pp. 2087–2130.

[89] LANG, S. 2002. *Algebra, 3rd ed.* New York : Springer, 2002. ISBN 0-387-95385-X.

[90] LEISS, E. L. 1999. *Language Equations*. New York : Springer, 1999. ISBN 978-1-4612-7436-0.

[91] MÄKINEN, E. 1984. On Context-Free and Szilard Languages. In *BIT Numerical Mathematics*. 1984, vol. 24, no. 2, pp. 164–170.

[92] MÄKINEN, E. 1985. A Note on Depth-First Derivations. In *BIT Numerical Mathematics*. 1985, vol. 25, no. 1, pp. 293–296.

[93] MÄKINEN, E. 1998. Ranking and Unranking Left Szilard Languages. In *International Journal of Computer Mathematics*. 1998, vol. 68, no. 1–2, pp. 29–38.

[94] MEDVEĎ, M. 2000. *Dynamické systémy*. Bratislava : Univerzita Komenského v Bratislave, 2000. ISBN 80-223-1346-7.

[95] MEINECKE, I. 2011. Valuations of Weighted Automata: Doing It in a Rational Way. In *Algebraic Foundations in Computer Science*. Heidelberg : Springer, 2011, pp. 309–346.

[96] MINC, H. 1988. *Nonnegative Matrices*. New York : John Wiley & Sons, 1988. ISBN 0-471-83966-3.

[97] MORIYA, E. 1973. Associate Languages and Derivational Complexity of Formal Grammars and Languages. In *Information and Control*. 1973, vol. 22, no. 2, pp. 139–162.

[98] PAPADIMITRIOU, C. H. 1994. *Computational Complexity*. Reading : Addison-Wesley, 1994. ISBN 0-201-53082-1.

[99] PEMANTLE, R. – WILSON, M. C. 2013. *Analytic Combinatorics in Several Variables*. Cambridge : Cambridge University Press, 2013. ISBN 978-1-107-03157-9.

[100] PENTTONEN, M. 1974. On Derivation Languages Corresponding to Context-Free Grammars. In *Acta Informatica*. 1974, vol. 3, pp. 285–291.

[101] PETRE, I. – SALOMAA, A. 2009. Algebraic Systems and Pushdown Automata. In *Handbook of Weighted Automata*. Heidelberg : Springer, 2009, pp. 257–289.

[102] POINCARÉ, H. 1890. Sur le problème des trois corps et les équations de la dynamique. In *Acta Mathematica*. 1890, vol. 13, pp. 1–270.

[103] PORAT, S. et al. 1982. Fair Derivations in Context-Free Grammars. In *Information and Control*. 1982, vol. 55, no. 1–3, pp. 108–116.

[104] POWELL, M. D. – SCHUCHMAN, E. – VIJAYKUMAR, T. N. 2005. Balancing Resource Utilization to Mitigate Power Density in Processor Pipelines. In *Proceedings of the 38th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO-38)*. Washington : IEEE Computer Society, 2005, pp. 294–304.

[105] PRIESTLEY, H. A. 2003. *Introduction to Complex Analysis, 2nd ed.* Oxford : Oxford University Press, 2003. ISBN 0-19-852562-1.

[106] ROVAN, B. 1981. A Framework for Studying Grammars. In *Mathematical Foundations of Computer Science (MFCS 1981)*. Heidelberg : Springer, 1981, pp. 473–482.

[107] ROZENBERG, G. – SALOMAA, A. 1980. *The Mathematical Theory of L Systems*. New York : Academic Press, 1980. ISBN 0-12-597140-0.

[108] SAKAROVITCH, J. 2009. *Elements of Automata Theory*. Cambridge : Cambridge University Press, 2009. ISBN 978-0-521-84425-3.

[109] SALOMAA, A. 1973. *Formal Languages*. New York : Academic Press, 1973. ISBN 0-12-615750-2.

[110] SALOMAA, A. – SOITTOLA, M. 1978. *Automata-Theoretic Aspects of Formal Power Series*. New York : Springer, 1978. ISBN 0-387-90282-1.

[111] SARKAR, P. 2001. Pushdown Automaton with the Ability to Flip its Stack. In *Electronic Colloquium on Computational Complexity (ECCC)*, 2001, Report No. 8.

[112] SCHÜTZENBERGER, M. P. 1963. On Context-Free Languages and Pushdown Automata. In *Information and Control*. 1963, vol. 6, no. 3, pp. 246–264.

[113] SIMMONS, G. F. 1963. *Introduction to Topology and Modern Analysis*. New York : McGraw-Hill, 1963. ISBN 0-07-059784-7.

[114] STANLEY, R. P. 1997. *Enumerative Combinatorics, Vol. 1*. Cambridge : Cambridge University Press, 1997. ISBN 0-521-55309-1.

[115] STANLEY, R. P. 1999. *Enumerative Combinatorics, Vol. 2*. Cambridge : Cambridge University Press, 1999. ISBN 0-521-56069-1.

[116] TITCHMARSH, E. C. 1939. *The Theory of Functions, 2nd ed.* Oxford : Oxford University Press, 1939. ISBN 0-19-853349-7.

[117] TRETMANS, J. 2008. Model Based Testing with Labelled Transition Systems. In *Formal Methods and Testing: An Outcome of the FORTEST Network*. Heidelberg : Springer, 2008, pp. 1–38.

[118] VAN EMDE BOAS, P. 1990. Machine Models and Simulations. In *Handbook of Theoretical Computer Science. Volume A: Algorithms and Complexity*. Amsterdam : Elsevier, 1990, pp. 1–66.

[119] VARGA, R. S. 2000. *Matrix Iterative Analysis*. Heidelberg : Springer, 2000. ISBN 978-3-540-66321-8.

[120] WAGNER, K. – WECHSUNG, G. 1986. *Computational Complexity*. Heidelberg : Springer, 1986. ISBN 978-90-277-2146-4.

[121] WECHSUNG, G. 1975. Characterization of Some Classes of Context-Free Languages in Terms of Complexity Classes. In *Mathematical Foundations of Computer Science (MFCS 1975)*. Heidelberg : Springer, 1975, pp. 457–461.

[122] WECHSUNG, G. 1979. A Crossing Measure for 2-Tape Turing Machines. In *Mathematical Foundations of Computer Science (MFCS 1979)*. Heidelberg : Springer, 1979, pp. 508–516.

[123] WIELANDT, H. 1950. Unzerlegbare, nicht negative Matrizen. In *Mathematische Zeitschrift*. 1950, vol. 52, no. 1, pp. 642–648.

[124] WILF, H. S. 1994. *Generatingfunctionology, 2nd ed.* San Diego : Academic Press, 1994. ISBN 978-0-08-057151-5.

[125] WOODS, A. R. 1997. Coloring Rules for Finite Trees, and Probabilities of Monadic Second Order Sentences. In *Random Structures & Algorithms*. 1997, vol. 10, no. 4, pp. 453–485.

[126] XU, C. – LAU, F. C. M. 1997. *Load Balancing in Parallel Computers: Theory and Practice*. Boston : Kluwer Academic Publishers, 1997. ISBN 978-0-585-27256-6.

[127] ZLATOŠ, P. 2011. *Lineárna algebra a geometria*. Bratislava : Albert Marenčin PT, 2011. ISBN 978-80-8114-111-9.