A Circuit Complexity Approach to Transductions*

Michaël Cadilhac[1], Andreas Krebs[1], Michael Ludwig[1], and Charles Paperman[2]

[1] Wilhelm Schickard Institut, Universität Tübingen
[2] University of Warsaw

Abstract. We investigate the deterministic rational transductions computable by constant-depth, polysize circuits. To this end, we first propose a framework of independent interest to express functions of variable output length using circuits, and argue for its pertinence. We then provide a general characterization of the set of transductions realizable by such circuits, relying on a notion of continuity. We deduce that it is decidable whether a transduction is definable in AC^0 and, assuming a well-established conjecture, the same for ACC^0.

Automata provide for an elegant and intuitive way to express regular languages, and even their intrinsic sequentiality is no obstacle to implementation, as they admit efficient parallelization. The connections between the theory of regular languages and that of circuits were first unveiled in the 1960s—it is now folklore that any regular language admits a (highly uniform) family of logdepth, polysize, and constant fan-in circuits. The natural further steps taken up by McNaughton and Papert [8], and then by Barrington *et al.* [3] led to the characterization of the regular languages in the class AC^0 of constant-depth, polysize, unbounded fan-in circuits using a decidable invariant. The property therein deviates sharply from the prevailing line of work at the time, which relied on the study of the syntactic monoid of regular languages. It was indeed known from the lower bound of Furst, Saxe and Sipser [6] that the language PARITY $= \{w \in \{0,1\}^* \mid |w|_1 \equiv 0 \bmod 2\}$ is not in AC^0, while the language EVEN of even-length words over $\{0,1\}$, which has the same syntactic monoid, does belong to AC^0. Hence the class of regular languages in AC^0 does not admit a characterization solely in terms of the syntactic monoid.

We propose to take this study to the functional case, that is, to characterize the functions realized by rational transducers that are expressible by an AC^0 circuit family. Similarly to the context at the time of [3], we face a situation where, to the best of our knowledge, most characterizations focused on algebraic properties that would blur the line between PARITY and EVEN. This is the case, e.g., for the characterization of functional aperiodic nondeterministic transducers by FO-translations [7].

* An extended version of this abstract with the same title has been submitted to MFCS 2015.

We rely on a property we abusively call *continuity* for a class $\mathcal{V}$, as borrowed from the field of topology: a transduction is $\mathcal{V}$-continuous if it preserves $\mathcal{V}$ by inverse image. It is well known that any transduction τ is continuous for the regular languages; together with an additional property on the output length of τ, this even characterizes deterministic transductions [5]. Namely, with $d(u,v) = |u| + |v| - |u \wedge v|$, where $u \wedge v$ is the largest common prefix of u and v, the latter property is that $d(\tau(u), \tau(v)) \leq k \times d(u,v)$, a strong form of *uniform* continuity. Continuity thus appears as a natural invariant when characterizing transductions—the *forward* behaviors of τ, that is, its images, are less relevant, as any NP problem is the image of Σ^* under an AC^0 function [4].

Our contributions are three-fold:

- We propose a model of circuits that allows for functions of unrestricted output length: as opposed to previous models, e.g., [11], we do not impose the existence of a mapping between the input and output lengths.
- Relying on this model, we characterize the deterministic rational transductions in $\mathrm{AC}^0(\mathcal{V})$—that is, AC^0 with gates for the languages in a well-structured class $\mathcal{V}$. Our characterization relies for one part on algebraic objects similar to the ones used in [3], through the use of the modern framework of *lm-varieties* [10]. For the other part, we rely on the notion of *continuity*. This bears a striking resemblance to the characterization of Reutenaeur and Schützenberger [9] of the transductions with a group as transition monoid.
- Our characterization then leads to the decidability of the membership of a deterministic rational transduction in AC^0 or in ACC^0. The procedure is effective, in the sense that an appropriate circuit can be produced realizing the transduction.

1 Circuit frameworks for variable-length functions

In the literature, most of the work on functions computed by circuits focus on variants of the class FAC^0 (see, e.g., [1]). In these, multiple (ordered) output gates are provided, and there is thus an implicit mapping from input length to output length. Towards circumventing this limitation, we propose a few different frameworks, and establish some formal shortcomings in order to legitimize our final choice. Our main consideration is that a function defined using a constant-depth, polysize circuit family of some kind should be AC^0-continuous—this corresponds to a simple composition of the circuits. In particular, any FAC^0 function is readily seen to be AC^0-continuous.

Recent research on the circuit complexity of variable-length functions [2] simply disregard this issue by allowing for a mechanism of *deactivation* of outputs. When considering expressive enough circuit models (e.g., NC^1), this is of no concern: the deactivated gates can be arbitrary moved to the end of the output, so that the output is presented in a natural way. However, lower complexity classes, such as AC^0, are not able to perform such a sorting. More precisely, we show:

Proposition 1. *There is a transduction expressible as a constant-depth, polysize family of deactivating circuits which is not* AC^0*-continuous.*

Another natural approach is to give (u, v) in input to the circuit, and have the circuit output 1 iff $v = f(u)$. We also show that such a definition would not respect AC^0-continuity. We are thus led to introduce:

Definition 1 (Functional circuits). *A function $\tau \colon \Sigma^* \to T^*$ is expressed as a circuit family $(C_m^n)_{n,m\geq 0}$, where C_m^n is a circuit with n inputs and $m+1$ outputs, if:*

$$(\forall u, v \in \Sigma^*) \qquad \tau(u) = v \Leftrightarrow C_{|v|}^{|u|}(u) = (v, 1) \ .$$

The size *of the family is the mapping from $\mathbb{N}$ to $\mathbb{N} \cup \{\infty\}$, defined by $n \mapsto \sup_{m\geq 0} |C_m^n|$. Similarly, the* depth *of the family is the mapping that associates n to the supremum of the depths of each C_m^n. The class $\mathrm{FAC}_\mathrm{v}^0$, standing for* functions in AC^0 with variable output length*, is the class of functions expressible as a family of constant-depth, polysize circuits. The class $\mathrm{FAC}_\mathrm{v}^0(\mathcal{V})$ denotes such circuits with gates for the languages in the class $\mathcal{V}$, and we let $\mathrm{FACC}_\mathrm{v}^0 = \mathrm{FAC}_\mathrm{v}^0(\mathrm{MOD})$ for MOD the modulo gates.*

2 The transductions in $\mathbf{FAC_v^0}(\mathcal{V})$

In sharp contrast with the work of Reutenaeur and Schützenberger [9], we are especially interested in the shape of the outputs of the transduction. It turns out that most of its complexity is given by the following output-length function. Let $\mathsf{MinT}(\tau)$, for a transduction τ, be a minimal transducer for τ.

Definition 2 ($\tau_\#$). *Let τ be a transduction. The function $\tau_\# \colon \Sigma^* \to \mathbb{N}$ is the output-length function of $\mathsf{MinT}(\tau)$ with all the states deemed final. In other words, $\tau_\#(s)$ is the length of the output produced while reading w from the initial state in $\mathsf{MinT}(\tau)$.*

Further, we say that a transducer is $\mathcal{V}$-all-definable, for a class $\mathcal{V}$ of languages, if for all of its states q, the language of the underlying automaton with q as the only final state is in $\mathcal{V}$. We show that this language-theoretic concept is equivalent in some precise way to the modern framework of lm-varieties [10]. This enables a study that stands in the algebraic tradition with no appeal to its tools. We can then state:

Theorem 1. *Let τ be a transduction and $\mathcal{V}$ be such that $\mathrm{AC}^0(\mathcal{V}) \cap \mathrm{REG} = \mathcal{V}$. The following constitutes a chain of implications:*
(i) $\tau \in \mathrm{FAC}_\mathrm{v}^0(\mathcal{V})$*;*
(ii) τ *is* $\mathrm{AC}^0(\mathcal{V})$*-continuous;*
(iii) τ *is* $\mathcal{V}$*-continuous;*
(iv) $\mathsf{MinT}(\tau)$ *is* $\mathcal{V}$*-all-definable.*
Moreover, if $\tau_\# \in \mathrm{FAC}_\mathrm{v}^0(\mathcal{V})$ then (iv) implies (i). Somewhat conversely, (i) implies $\tau_\# \in \mathrm{FAC}_\mathrm{v}^0(\mathcal{V})$.

In the case of $\mathrm{FAC}_\mathrm{v}^0$ and $\mathrm{FACC}_\mathrm{v}^0$, we are able to express Theorem 1 using a syntactical restriction on the transducer for τ:

Definition 3 (Constant ratio). *A transducer has* constant ratio *if every two words of the same length looping on a state produce outputs of the same length from this state. In symbols, for any state q and any words u, v of the same length, $\delta(q,u) = \delta(q,v) = q$ implies $|\nu(q,u)| = |\nu(q,v)|$.*

Writing $\mathcal{QA}$ and $\mathcal{M}_{\text{sol}}$ for the regular languages in AC^0 and ACC^0, respectively, we can then state:

Corollary 1. *Let τ be a transduction. The following are equivalent, where the the "resp." part assumes $\text{ACC}^0 \neq \text{TC}^0$:*
(i) $\tau \in \text{FAC}^0_{\text{v}}$ *(resp.* $\in \text{FACC}^0_{\text{v}}$*);*
(ii) τ *is continuous for* AC^0 *(resp. for* ACC^0*) and* $\mathsf{MinT}(\tau)$ *has constant ratio;*
(iii) τ *is continuous for* $\mathcal{QA}$ *(resp. for* $\mathcal{M}_{sol}$*) and* $\mathsf{MinT}(\tau)$ *has constant ratio;*
(iv) $\mathsf{MinT}(\tau)$ *is all-definable for* $\mathcal{QA}$ *(resp. for* $\mathcal{M}_{sol}$*) and has constant ratio.*

Moreover, the latter property can be tested effectively, hence:

Theorem 2. *It is decidable whether a transducer realizes an* FAC^0_{v} *function. If it does, then a circuit family can be constructed. The same holds for* FACC^0_{v} *assuming* $\text{ACC}^0 \neq \text{TC}^0$*.*

References

1. Agrawal, M., Allender, E., Datta, S.: On TC^0, AC^0, and arithmetic circuits. J. Comput. Syst. Sci 60(2), 395–421 (2000)
2. Allender, E., Mertz, I.: Complexity of regular functions. In: Language and Automata Theory and Applications. pp. 449–460 (2015), `http://dx.doi.org/10.1007/978-3-319-15579-1_35`
3. Barrington, D.A.M., Compton, K., Straubing, H., Thérien, D.: Regular languages in NC^1. J. Computer and System Sciences 44(3), 478–499 (1992)
4. Beyersdorff, O., Datta, S., Krebs, A., Mahajan, M., Scharfenberger-Fabian, G., Sreenivasaiah, K., Thomas, M., Vollmer, H.: Verifying proofs in constant depth. TOCT 5(1), 2 (2013)
5. Choffrut, C.: A generalization of Ginsburg and Rose's characterization of G-S-M mappings. In: ICALP. pp. 88–103. Springer-Verlag, London, UK, UK (1979)
6. Furst, M., Saxe, J.B., Sipser, M.: Parity, circuits, and the polynomial-time hierarchy. Theory of Computing Systems 17, 13–27 (1984)
7. Lautemann, C., McKenzie, P., Schwentick, T., Vollmer, H.: The descriptive complexity approach to LOGCFL. J. Computer and Systems Sciences 62(4), 629–652 (2001)
8. McNaughton, R., Papert, S.: Counter-Free Automata. The MIT Press, Cambridge, Mass. (1971)
9. Reutenaeur, C., Schützenberger, M.P.: Variétés et fonctions rationnelles. Theoretical Computer Science 145(1–2), 229–240 (Jul 1995)
10. Straubing, H.: On logical descriptions of regular languages. In: Rajsbaum, S. (ed.) LATIN. LNCS, vol. 2286, pp. 528–538. Springer (2002)
11. Vollmer, H.: Introduction to circuit complexity. Springer-Verlag, Berlin-Heidelberg-New York-Barcelona-Hong Kong-London-Milan-Paris-Singapur-Tokyo (1999)