



## A note on monotone real circuits <sup>☆</sup>



Pavel Hrubeš\*, Pavel Pudlák

Institute of Mathematics of the Czech Academy of Sciences, Prague, Czech Republic

### ARTICLE INFO

#### Article history:

Received 13 April 2017  
 Received in revised form 4 October 2017  
 Accepted 12 November 2017  
 Available online 14 November 2017  
 Communicated by L. Viganò

#### Keywords:

Computational complexity  
 Monotone real circuit  
 Karchmer–Wigderson game

### ABSTRACT

We show that if a Boolean function  $f : \{0, 1\}^n \rightarrow \{0, 1\}$  can be computed by a monotone real circuit of size  $s$  using  $k$ -ary monotone gates then  $f$  can be computed by a monotone real circuit of size  $O(sn^{k-2})$  which uses unary or binary monotone gates only. This partially solves an open problem presented in [2]. In fact, in size  $O(sn^{k-1})$ , the circuit uses only unary monotone gates and binary addition. We also show that if the monotone Karchmer–Wigderson game of  $f$  can be solved by a “real communication protocol” of size  $s$  then  $f$  can be computed by a monotone real circuit of the same size.

© 2017 Published by Elsevier B.V.

### 1. Introduction

In this note, we present some structural properties of the computational model of *monotone real circuits*. Motivated by proof complexity applications, monotone real circuits were introduced in [10] where an exponential lower bound was obtained for this model; a similar lower bound was independently obtained in [3]. The first issue we address here is the arity of gates used in computation. Monotone real circuits, in their usual definition, use binary (and unary) gates over the reals. But if we allow, say, ternary gates, can we significantly speed-up the computation? For Boolean circuits, or circuits involving functions over a *finite* domain, the answer is obvious: every ternary gate can be expressed as a composition of a constant number of binary gates, so using ternary gates can give an advantage of at most a constant factor. In the case of computations over the reals, or any *infinite* ordered set, an analogous statement is far from obvious. In [2], the possibility of

gate-by-gate by simulation was stated as an open problem. We don't know how to solve this problem, but we nevertheless show that ternary gates, or gates of not too large an arity, indeed cannot substantially speed-up real computations. We also show that the only gates needed are monotone unary gates and binary additions. This is interesting in comparison with Kolmogorov's superposition theorem [1,8] which states that every  $k$ -ary continuous function can be expressed using unary continuous functions and addition.

Second, we give a correspondence between monotone real circuits and real communication protocols. This resembles a similar correspondence between Boolean circuits and PLS-based protocols of Razborov [12], and, more closely, the construction of Krajíček [9] (see also [11,13]). The motivation is the following. In [7], Karchmer and Wigderson characterized Boolean circuit depth in terms of deterministic complexity of certain communication games. This game-theoretic viewpoint has been very useful both in proving lower bounds and constructing upper bounds on circuit *depth*. The constructions in [12,9] are intended to give a similar interpretation of Boolean circuit size in terms of games. This is useful especially when proving feasible interpolation theorems in various proof systems. We now give a similar characterization of monotone real circuits in terms of real games. As in [9] or [11], such a characteriza-

<sup>☆</sup> The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) / ERC grant FEALORA 339691.

\* Corresponding author.

E-mail addresses: pahrubes@gmail.com (P. Hrubeš),  
 pudlak@math.cas.cz (P. Pudlák).

tion can be directly used to give a simple proof of feasible interpolation in the Cutting Planes proof system, or applied to the related notion of “unsatisfiability certificates” introduced in [6].

## 2. Monotone real circuits of high fan-in

For  $x, y \in \mathbb{R}^n$ , we write  $x \leq y$  if  $x_i \leq y_i$  for every coordinate  $i \in [n] = \{1, \dots, n\}$ . A function  $f : S \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$  will be called *monotone*, if for every  $x, y \in S$ ,  $x \leq y$  implies  $f(x) \leq f(y)$ . We remark that, assuming  $\sup\{|f(x)| : x \in S\} < \infty$ ,  $f$  is monotone iff it is a restriction of some *total* monotone function  $g : \mathbb{R}^n \rightarrow \mathbb{R}$ . Since we will usually deal with a finite  $S$ , the distinction between partial/total functions is hence quite unimportant.<sup>1</sup>

A  $k$ -ary monotone real circuit is a finite directed acyclic graph with every node of in-degree at most  $k$ . It has one output node of out-degree zero. Nodes of in-degree zero are called input nodes and are labeled with variables. Every other node  $v$  of in-degree  $p$  is labeled with a (total) monotone function  $g_v : \mathbb{R}^p \rightarrow \mathbb{R}$ . The size of the circuit is the number of its gates. The circuit computes a function  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  in the obvious way: an input label- $led$  with  $x_i$  computes  $x_i$ , otherwise a node label- $led$  with  $g_v$  computes  $g_v(f_1, \dots, f_p)$  where  $f_1, \dots, f_p$  are the functions computed by its predecessors. We say that the circuit computes a partial function, if the output node computes some extension of it.

Our main result is:

**Theorem 1.** Assume that  $f : \{0, 1\}^n \rightarrow \{0, 1\}$  can be computed by a  $k$ -ary monotone real circuit of size  $s$ . Then  $f$  can be computed by a binary monotone real circuit of size  $O(sn^{k-2})$ . Moreover, increasing the size to  $O(sn^{k-1})$ , we can assume that the circuit uses only unary gates and binary additions.

Here, *binary addition* is a fan-in 2 gate that adds the two real numbers. In the rest of this section, we prove [Theorem 1](#). The main ingredient, [Theorem 4](#), is to show that a monotone function on a finite set  $S \subseteq \mathbb{R}^k$  can be computed by a binary monotone real circuit of size  $O(\log_2(|S|)^{k-2})$ . To conclude [Theorem 1](#), we then note that in a monotone real circuit computing  $f : \{0, 1\}^n \rightarrow \{0, 1\}$ , the intermediary gates can be restricted to domain of size at most  $2^n$ . [Theorem 4](#) itself is proved using two technical lemmas. [Lemma 2](#) in its simplest, but already nontrivial, form states the following: a monotone two-valued function  $f : S_0 \rightarrow \{0, 1\}$ , with  $S_0 \subseteq \mathbb{R}^2$  finite, can be computed using one binary addition and unary monotone gates. The lemma is then extended to talk about an arbitrary  $k$ , as well as about a finite union of two-valued functions, under the assumption that their domains and ranges are appropriately ordered. In [Lemma 3](#), we take a monotone  $k$ -ary function  $f$  and “stretch” its domain so that [Lemma 2](#) becomes applicable. This allows to compute  $f$  using functions of arity  $k - 1$  and proceed inductively.

**Lemma 2.** Let  $S_0, \dots, S_m$  be finite subsets of  $\mathbb{R}^k$ ,  $k \geq 1$ , such that  $S_i \subseteq (i, i + 1)^k$  for every  $i$  and let  $S := \bigcup_{i=0}^m S_i$ . Assume that  $f : S \rightarrow \mathbb{R}$  is a monotone function such that  $f(S_i) \subseteq \{2i, 2i + 1\}$  for every  $i$ . Then there exist monotone functions  $g, h$  such that for all  $\langle x_1, \dots, x_k \rangle \in S$ ,

$$f(x_1, \dots, x_k) = g(x_1 + h(x_2, \dots, x_k)).$$

**Proof.** We identify  $\mathbb{R} \times \mathbb{R}^{k-1}$  with  $\mathbb{R}^k$ . Let  $X_i := \{x \in \mathbb{R} : \exists y \in \mathbb{R}^{k-1}, \langle x, y \rangle \in S_i\}$  and  $Y_i := \{y \in \mathbb{R}^{k-1} : \exists x \in \mathbb{R}, \langle x, y \rangle \in S_i\}$  be the projections of  $S_i$  to first coordinate, and the last  $k - 1$  coordinates, respectively. Let  $Y := \bigcup_{i=0}^m Y_i$ . For  $y \in Y$ , let

$$\alpha(y) := \min(\{x \in X_i : f(x, y) = 2i + 1\} \cup \{i + 1\}),$$

$$\text{if } y \in Y_i.$$

This guarantees that, for every  $\langle x, y \rangle \in S_i$ ,  $f(x, y) = 2i + 1$  iff  $x \geq \alpha(y)$ . Since  $|x - \alpha(y)| < 1$ , we can write

$$f(x, y) = \lfloor (2i + 1 + x - \alpha(y)) \rfloor, \text{ for every } \langle x, y \rangle \in S_i.$$

Since  $f$  was a monotone function,  $-\alpha$  is a monotone function on each of the sets  $Y_i$ . Define

$$h(y) := 2i + 1 - \alpha(y), \text{ if } y \in Y_i.$$

The function is monotone on every set  $Y_i$ . Moreover, since  $h(S_i) \subseteq [i, i + 1]$ ,  $h$  is monotone on the whole of  $Y$ . This gives the expression  $f(x, y) = \lfloor x + h(y) \rfloor$  as required.  $\square$

**Lemma 3.** Let  $f : S \rightarrow T$  be a monotone function where  $S \subseteq \mathbb{R}^k$  and  $T \subseteq \mathbb{R}$  are finite sets. Let  $t := \lceil \log_2 |T| \rceil$ . Then  $f$  can be computed by a monotone real circuit of size  $O(k(t - 1))$  such that the circuit uses only unary gates, binary addition gates, and  $t$  gates of arity  $k - 1$ .

**Proof.** W.l.o.g., we can assume that  $S \subseteq (0, 1)^k$  and  $T = \{0, 1, \dots, 2^t - 1\}$ . The circuit is constructed by induction with respect to  $t$ . For  $t = 0$ , the function is constant. If  $t = 1$ , this is simply [Lemma 2](#) (with  $m = 0$ ). Assume that  $t > 1$  and let  $f' := \lfloor \frac{1}{2} f \rfloor$ . Hence,  $f' : S \rightarrow \{0, \dots, 2^{t-1} - 1\}$  and assume we have constructed a circuit for  $f'$ . For  $x = \langle x_1, \dots, x_k \rangle$ , define  $i + x := \langle i + x_1, \dots, i + x_k \rangle$ , and let  $S_i := \{i + x : x \in S\}$ . Define  $h : \bigcup_{i=0}^{2^{t-1}-1} S_i \rightarrow \{0, \dots, 2^t - 1\}$  by putting, for  $z \in S_i$  of the form  $z = i + x$ ,

$$h(z) := \begin{cases} 2i & \text{if } f(x) \leq 2i \\ 2i + 1 & \text{if } f(x) \geq 2i + 1. \end{cases}$$

The function is monotone on each of the sets  $S_i$ , and since  $h(S_i) \subseteq \{2i, 2i + 1\}$ , it is monotone on the whole  $\bigcup_{i=0}^{2^{t-1}-1} S_i$ . The definition guarantees that, for all  $x \in S$ ,

$$f(x) = h(f'(x) + x). \quad (1)$$

By [Lemma 2](#),  $h$  can be computed by a circuit with 3 non-input gates, such that the circuit uses one binary addition, one unary gate, and one gate of arity  $k - 1$ . Hence, by (1),  $f$  can be computed from  $f'$  using one additional  $(k - 1)$ -ary gate and  $k + 2$  binary addition/unary gates. This gives the required circuit for  $f$ .  $\square$

<sup>1</sup> Also, the assumption  $\sup\{|f(x)| : x \in S\} < \infty$  could be removed, had we decided to work over  $\mathbb{R} \cup \{\pm\infty\}$ .

Download English Version:

<https://daneshyari.com/en/article/6874234>

Download Persian Version:

<https://daneshyari.com/article/6874234>

[Daneshyari.com](https://daneshyari.com)