A note on monotone real circuits

Pavel Hrubeš and Pavel Pudlák [*]

March 14, 2017

Abstract

We show that if a Boolean function $f : \{0,1\}^n \to \{0,1\}$ 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-Wigerson 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.

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 [8] 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, 6] which states that every k-ary continuous function can be expressed using unary continuous functions and addition.

[*] Institute of Mathematics of AVCR, Prague, pahrubes@gmail.com, pudlak@math.cas.cz. Supported by ERC grant FEALORA 339691.

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 [10], and, more closely, the construction of Krajíček [7] (see also [9, 11]). The motivation is the following. In [5], 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 [10, 7] 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 [7] or [9], such a characterization 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 [4].

2 Simulation of monotone real circuits of higher fan-in by circuits with fan-in 2

For $x, y \in \mathbb{R}^n$, we write $x \leq y$ if $x_i \leq y_i$ for every coordinate $i \in [n] = \{1, \ldots, n\}$. A function $f : S \subseteq \mathbb{R}^n \to \mathbb{R}$ 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 $g : \mathbb{R}^n \to \mathbb{R}$. Since we will usually deal with a finite S, the distinction between partial/total functions is hence quite unimportant.[1]

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 labelled with variables. Every other node v of in-degree p is labelled with a (total) monotone function $g_v : \mathbb{R}^p \to \mathbb{R}$. The size of the circuit is the number of its gates. The circuit computes a function $f : \mathbb{R}^n \to \mathbb{R}$ in the obvious way: an input labelled with x_i computes x_i, otherwise a node labelled with g_v computes $g_v(f_1, \ldots, f_p)$ where $f_1, \ldots, f_p$ 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 $f : \{0,1\}^n \to \{0,1\}$ 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.

Lemma 2. *Let $S_0, \ldots, S_m$ 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 $f : S \to \mathbb{R}$ is a monotone function such that*

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

$f(S_i) \subseteq \{2i, 2i+1\}$ for every i. Then there exist monotone functions g, h such that for all $\langle x_1, \dots, x_k\rangle \in S$,

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

Note that even the case $m = 0$ and $k = 2$ is interesting: a monotone $f : S' \times S' \to \{0, 1\}$, $S' \subseteq \mathbb{R}$ finite, can be computed using one binary addition and unary monotone gates.

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

$$\alpha(y) := \min\left(\{x \in X_i : f(x, y) = 2i+1\} \cup \{i+1\}\right) \,,\ \text{if } y \in Y_i.$$

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

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

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. □

Lemma 3. *Let $f : S \to T$ 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, $f' : S \to \{0, \dots, 2^{t-1} - 1\}$ and assume we have constructed a circuit for f'. For $z = \langle x_1, \dots, x_k\rangle$, define $i + z := \langle i + x_1, \dots, i + x_k\rangle$, and let $S_i := \{i + z : z \in S\}$. Define $h : \bigcup_{i=0}^{2^{t-1}-1} S_i \to \{0, \dots, 2^{t+1} - 1\}$ by putting, for $z \in S_i$ of the form $z = i + z'$,

$$h(z) := \begin{array}{ll} 2i & \text{if } f(z') \leq 2i \\ 2i+1 & \text{if } f(z') \geq 2i+1 \,. \end{array}$$

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 $z' \in S$,

$$f(z') = h(f'(z') + z'). \tag{1}$$

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. □

Theorem 4. *Let $f : S \to \mathbb{R}$ be a monotone function where $S \subseteq \mathbb{R}^k$ is a finite set. Then f can be computed by a binary monotone real circuit of size $O(\log_2(|S|)^{k-2})$. Moreover, increasing the size to $O(\log_2(|S|)^{k-1})$, we can assume that the circuit uses only unary gates and binary additions.*

Proof. We prove the "moreover" part, the rest is similar. Let $\lambda(k, m)$ denote the minimum s, such that for every $S \subseteq \mathbb{R}^k$ with $|S| \leq 2^m$ and every monotone function $f : S \to \mathbb{R}$, f can be computed by a circuit of size $\leq s$ with only unary gates and binary additions. If $m \leq 1$, it is easy to show that $\lambda(k, m) = 2$. (For then S consists of at most two points x, y. If f is non-constant, we can assume $f(x) < f(y)$, hence for some coordinate $x_i < y_i$, and we can write f as a function of this coordinate only.) Hence we assume $m \geq 2$.

If $k = 1$, $\lambda(k, m) = 2$. If $k > 1$, the previous lemma gives

$$\lambda(k, m) \leq ckm + m\lambda(k-1, m)\,, \tag{2}$$

for a suitable constant $c \geq 2$. This is seen as follows: given $f : S \to \mathbb{R}$, its range has size at most $|S|$. The lemma then shows that f can be computed using cmk additions, unary gates, and m gates of arity $k - 1$. The latter gates can be restricted to domain of size at most $|S|$, as only that many values can appear in the computation of f. (2) has a solution $\lambda(k, m) \leq c_1 m^{k-1} - c_2(k + 1)$ with $c_2 > 0$ a sufficiently large constant. Setting $c_1 > 0$ to satisfy the initial condition $\lambda(1, m) = 2$ gives $\lambda(k, m) \leq c_1 m^{k-1}$. □

Proof of Theorem 1. Given a monotone real circuit computing $f : \{0, 1\}^n \to \{0, 1\}$, we can assume that every gate in the circuit has domain of size at most $|\{0, 1\}^n| = 2^n$, and apply the previous theorem. □

3 Simulation of monotone real protocols by circuits

The following is a modification of a concept defined by Krajíček in [7], Definition 2.1.

Definition. *A* real protocol of degree k *is a directed acyclic graph $G = (V, E)$ and a set of functions $r_v^0, r_v^1 : \{0, 1\}^n \to \mathbb{R}$ for every $v \in V$, such that*

(i). G has one source v_0 (a node of in-degree zero) and the out-degree of every vertex is at most k,

(ii). for every sink ℓ (a node of out-degree zero) there exists a variable x_i with $r_\ell^0 = r_\ell^1 = x_i$.

Let f be a partial monotone Boolean function in n variables. We say that the protocol solves the monotone KW game for f *(or simply* solves f*), if for every $x \in f^{-1}(0)$ and $y \in f^{-1}(1)$,*

(a) $r_{v_0}^0(x) < r_{v_0}^1(y)$,

(b) for every $v \in V$ with $p \geq 1$ children $u_1, \ldots, u_p$, if $r_v^0(x) < r_v^1(y)$ then there exists u_i with $r_{u_i}^0(x) < r_{u_i}^1(y)$.

The size *of a protocol is the number of vertices.*[2]

Let us motivate the definition. Recall the monotone Karchmer-Wigderson game for f: Player I has input x such that $f(x) = 0$ and Player II an input y with $f(y) = 1$. They are supposed to agree on some bit s.t. $x_i < y_i$. We say that a vertex v *is feasible for* x, y, if $r^0_v(x) < r^1_v(y)$ holds. Condition (a) says that v_0 is feasible for every $x \in f^{-1}(0)$, $y \in f^{-1}(1)$. Condition (b) says that if v is feasible for x, y then at least one of its children is feasible for x, y. Given $x \in f^{-1}(0)$ and $y \in f^{-1}(1)$, we can traverse the graph from the source to a sink ℓ using only vertices feasible for x, y. Since ℓ is feasible, we have $x_i = r^0_\ell(x) < r^1_\ell(y) = y_i$, hence $x_i < y_i$. In this sense, the protocol solves the KW game for f.

Given a k-ary monotone real circuit computing f, we can convert it to a protocol of degree k solving f: the underlying graph will be the same (with reversed direction of edges) and both functions r^0_v, r^1_v can be set equal to f_v, the function computable by the subcircuit rooted at v. To show that a protocol can be converted to a circuit is less obvious and is the essence of the next theorem.

Theorem 5. *Given a real protocol P solving f, there exists a monotone real circuit C computing f such that the underlying graph of C is the same as the graph of P (with reversed direction of edges).*

Proof. Assume that f and a protocol P are as above. Suppose w.l.o.g. that $r^0_v(z), r^1_v(z) \geq 0$ for all $v \in V$ and $z \in \{0,1\}^n$. For every $v \in V$, define a function $f_v : \{0,1\}^n \to \mathbb{R}$ inductively as follows. Given $z \in \{0,1\}^n$,

(i). if ℓ is a sink, $f_\ell(z) := r^0_\ell(z) = r^1_\ell(z)$,

(ii). for $v \in V$ with children $u_1, \ldots, u_p$,

$$f_v(z) := \max\{r^1_v(w) : w \in f^{-1}(1)\,,\ \forall_{i\in[p]} f_{u_i}(z) \geq r^1_{u_i}(w)\}\,, \tag{3}$$

where we put $f_v(z) := 0$ if the maximum is over the empty set.

We now claim that for every $x \in f^{-1}(0)$, $y \in f^{-1}(1)$ and $v \in V$

$$\begin{aligned} f_v(x) &\leq r^0_v(x)\,, && (4)\\ r^1_v(y) &\leq f_v(y)\,. && (5) \end{aligned}$$

This is proved by induction on depth. If v is a sink, both statements are clear. Suppose that v is as in (ii) and (4),(5) hold for the children of v. If $y \in f^{-1}(1)$, we can set $w := y$ in (3) to obtain $f_v(y) \geq r^1_v(y)$, proving (5). In order to prove (4), suppose that $x \in f^{-1}(0)$ and, for the sake of contradiction, $f_v(x) > r^0_v(x)$. Hence, (3) shows there exists $w \in f^{-1}(1)$ such that $r^1_v(w) > r^0_v(x)$ and for every u_i, $f_{u_i}(x) \geq r^1_{u_i}(w)$. On the the other hand, condition (b) demands that there exists some u_i with $r^0_{u_i}(x) < r^1_{u_i}(w)$. This gives $f_{u_i}(x) > r^0_{u_i}(x)$, contrary to the inductive assumption.

[2] Note that the functions r^0_v (r^1_v) need only be defined on $f^{-1}(0)$ (respectively on $f^{-1}(1)$).

To conclude the theorem, note that by (3), $f_v(z)$ can be computed from $f_{u_1}(z), \dots, f_{u_p}(z)$ by a single p-ary monotone gate. Hence, f_{v_0} can be computed by a k-ary monotone real circuit whose underlying graph is the same as that of P. Finally, (4), (5) and (a) give $f_{v_0}(x) < f_{v_0}(y)$ for every $x \in f^{-1}(0)$ and $y \in f^{-1}(1)$. Hence, we can compute f as

$$f(z) = \begin{cases} 0 & \text{if } f_{v_0}(z) < t \\ 1 & \text{if } f_{v_0}(z) \geq t \,, \end{cases}$$

where $t := \min_{y \in f^{-1}(1)} f_{v_0}(y)$ (this does not require an additional gate). □

Combined with Theorem 1, this implies.

Corollary 6. *Let f be an n-bit monotone Boolean function and let P be a real protocol for f of degree k solving f. Then f can be solved by*

(i). a protocol with the same graph as P such that for every $v \in V$, $r_v^0 = r_v^1$ is a monotone function,

(ii). a protocol of degree 2 and size $O(sn^{k-2})$.

In [7], Definition 1.1, Krajíček also introduced the following game. Two players communicate in rounds. In a given round of communication, Player I generates a real number a_0 and Player II a number a_1. They privately send the numbers to a referee, who in turn tells them whether $a_1 < a_2$ or not. Based on this information, the players proceed to the next round. Formally, the game is given by a binary tree T with two functions $s_v^0, s_v^1 : \{0,1\}^n \to \mathbb{R}$ for every non-leaf vertex v. From a vertex $v \in T$, the protocol continues to the left-hand child if $s_v^0(x) > s_v^1(y)$, and to the right-hand child otherwise.

Given a tree-like real protocol of degree 2 (according to our definition), we can easily transform it into a protocol of the type defined by Krajíček: for a non-leaf vertex v with the right-hand child u, put $s_v^0 := r_u^0$ and $s_v^1 := r_u^1$. However, we conjecture that the converse simulation without an essential increase of the depth is impossible.

4 Some observations

In [2], the following open problem was posed: over $\mathbb{R}$, can every monotone k-ary function be expressed as a composition of binary and unary monotone functions? Theorem 4 is a small step towards answering the question. Here are some relevant observations:

Proposition 7. *(i). There exists a monotone function $f : S \times S \to \mathbb{R}$ with $S \subseteq \mathbb{R}$ of size 3 such that f cannot be written as $f = g(h_1(x) + h_2(y))$ with g monotone (and h_1, h_2 arbitrary).*

(ii). There exists a monotone function $f : \mathbb{R}^2 \to \{0,1\}$ such that f cannot be written as a composition of unary monotone functions and addition over $\mathbb{R}$.

(iii). There exists a real closed field $\mathcal{R} \supseteq \mathbb{R}$ such that for every monotone $f : \mathbb{R}^2 \to \{0,1\}$, there exist monotone functions $g : \mathcal{R} \to \{0,1\}$, $h_1, h_2 : \mathbb{R} \to \mathcal{R}$ such that $f = g(h_1(x) + h_2(y))$.

In (iii), we extend the definition of monotone function to any ordered set, in an obvious way.

Proof. Part (ii). Recall that a function $g : \mathbb{R} \to \mathbb{R}$ is Borel measurable, if for every open set its preimage under g is a Borel set. We only need to know that every monotone/anti-monotone g is Borel measurable (because it is continuous everywhere, up to countably many points), and that Borel functions are closed under composition and addition. Pick $A \subseteq \mathbb{R}$ which is not a Borel set and let χ_A be its characteristic function. For $x, y \in \mathbb{R}$, define

$$f(x,y) := \begin{cases} 0 & \text{if } x+y<1 \\ \chi_A(x) & \text{if } x+y=1 \\ 1 & \text{if } x+y>1\,. \end{cases}$$

The function is monotone and $\chi_A(x) = f(x, 1-x)$. Hence, $f(x, 1-x)$ is not a Borel function and so f cannot be expressed as a composition of unary monotone functions and addition.

To prove (i) and (iii), we first make quite a general observation. Suppose that $\langle \mathcal{R}, 0, + \rangle$ is a linearly ordered Abelian group and that $f : S \times S \to \mathcal{R}$ is a monotone function with $S \subseteq \mathcal{R}$ possibly infinite. With f we associate the following set of linear inequalities $\mathcal{L}(f)$. For every a in S introduce new constant symbols α_a and β_a, and let

$$\mathcal{L}(f) := \{\alpha_{a_1} + \beta_{b_1} < \alpha_{a_2} + \beta_{b_2} : a_1, b_1, a_2, b_2 \in S\,,\ f(a_1, b_1) < f(a_2, b_2)\}\,.$$

Furthermore, let

$$\mathcal{L}_+(f) := \mathcal{L}(f) \cup \{\alpha_{a_1} \leq \alpha_{a_2}\,,\ \beta_{a_1} \leq \beta_{a_2} : a_1 \leq a_2 \in S\}\,.$$

It can be easily shown that:

Claim. *f can be written as $f = g(h_1(x) + h_2(y))$ with $g : \mathcal{R} \to \mathcal{R}$ monotone iff $\mathcal{L}(f)$ has a solution over $\mathcal{R}$. Similarly, for h_1, h_2 monotone and the system $\mathcal{L}_+(f)$.*

In order to prove (i), let $S := \{1, 2, 3\}$ and let the value of $f(i, j)$ be given by the i, j-entry of the matrix

$$\begin{pmatrix} 0 & 0 & 2 \\ 1 & 1 & 2 \\ 1 & 3 & 3 \end{pmatrix}.$$

Then f is monotone, whereas $\mathcal{L}(f)$ has no solution over $\mathbb{R}$ (or any ordered extension of $\mathbb{R}$). For, we can assume $\alpha_1, \beta_1 = 0$, and the system then contains inequalities

$$\beta_2 < \alpha_2\,,\ \alpha_3 < \beta_3\,,\ \alpha_2 + \beta_3 < \alpha_3 + \beta_2$$

with no solution.

For (iii), fix $f : \mathbb{R}^2 \to \{0, 1\}$ monotone. By Lemma 2 with $m = 0$, every finite subset of $\mathcal{L}_+(f)$ has a solution over $\mathbb{R}$. By compactness of first-order logic, $\mathcal{L}_+(f)$ has a solution over some extension $\mathcal{R}$ of $\mathbb{R}$. Here, we can impose any reasonable first-order properties on $\mathcal{R}$, as well as to achieve the statement for all monotone f simultaneously. □

Part (i) shows that in Lemma 2, some bound on range of f is necessary even for $m = 0$. Part (ii) shows that addition and unary monotone functions are not enough to capture general monotone functions over $\mathbb{R}$. Part (iii) is problematic in a different way: it indicates that the question "how can we express monotone functions in terms of simpler ones" may depend on which extension of $\mathbb{R}$ we have in mind.

References

[1] V. N. Arnold. On functions of three variables. *Doklady Akad. Nauk SSSR*, 114:679–681, 1957.

[2] Y. Filmus, P. Hrubeš, and M. Lauria. Semantic Versus Syntactic Cutting Planes. In *33rd Symposium on Theoretical Aspects of Computer Science (STACS 2016)*, volume 47 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 35:1–35:13, 2016.

[3] A. Haken and S. A. Cook. An exponential lower bound for the size of monotone real circuits. *J. Comput. Syst. Sci.*, 58(2):326–335, April 1999.

[4] P. Hrubeš and P. Pudlák. Random formulas, monotone circuits, and interpolation. *Electronic Colloquium in Computational Complexity*, 2017.

[5] M. Karchmer and A. Wigderson. Monotone circuits for connectivity require super-logarithmic depth. In *STOC*, 1988.

[6] A. N. Kolmogorov. On the representations of continuous functions of several variables by superpositions of continuous functions of fewer variables. *Doklady Akad. Nauk SSSR*, 108:179–182, 1956.

[7] J. Krajíček. Interpolation by a game. *Mathematical Logic Quarterly*, 44(4):450–458, 1998.

[8] P. Pudlák. Lower bounds for resolution and cutting plane proofs and monotone computations. *The Journal of Symbolic Logic*, 62(3):981–998, 1997.

[9] P. Pudlák. On extracting computations from propositional proofs. In *Proc. Annual Conference on Foundations of Software Technology and Theoretical Computer Science, Chennai*, Leibniz International Proceedings in Informatics, pages 30–41, 2010.

[10] A. A. Razborov. Unprovability of lower bounds on the circuit size in certain fragments of bounded arithmetic. *Izvestiya of the Russian Academy of Science*, 59(1):201–224, 1995.

[11] D. Sokolov. Dag-like communication and its applications. *Electronic Colloquium in Computational Complexity*, 2017.