The Complexity of Membership Problems for Circuits over Sets of Natural Numbers

Pierre McKenzie
Informatique et recherche opérationnelle
Université de Montréal
C.P. 6128, Succ. Centre-Ville, Montréal (Québec), H3C 3J7 Canada
mckenzie@iro.umontreal.ca
Klaus W. Wagner
Theoretische Informatik
Bayerische Julius-Maximilians-Universität Würzburg
Am Hubland, D-97074 Würzburg, Germany
wagner@informatik.uni-wuerzburg.de

Classification: Computational complexity

Abstract. The problem of testing membership in the subset of the natural numbers produced at the output gate of a $\{\cup, \cap, ^-, +, \times\}$ combinational circuit is shown to capture a wide range of complexity classes. Although the general problem remains open, the case $\{\cup, \cap, +, \times\}$ is shown NEXPTIME-complete, the cases $\{\cup, \cap, ^-, \times\}$, $\{\cup, \cap, \times\}$, $\{\cup, \cap, +\}$ are shown PSPACE-complete, the case $\{\cup, +\}$ is shown NP-complete, the case $\{\cap, +\}$ is shown $\mathrm{C_{=}L}$-complete, and several other cases are resolved. Interesting auxiliary problems are used, such as testing nonemptyness for union-intersection-concatenation circuits, and expressing each integer, drawn from a set given as input, as powers of relatively prime integers of one's choosing. Our results extend in nontrivial ways past work by Stockmeyer and Meyer (1973), Wagner (1984) and Yang (2000).

1 Introduction

Combinational circuits permeate complexity theory. Countless lower bounds, complexity class characterizations, and completeness results involve circuits over the boolean semiring (see [Vo00] among many others). Circuits and formulas over more general structures have been studied as well (see for a few examples [BCGR92, BM95, CMTV98, BMPT97, AJMV98, AAD00]).

In this paper, we study circuits operating on the natural numbers. These include 0 and are simply called the set $\mathbb{N}$ of *numbers* from now on. Next to the boolean semiring, the semiring of numbers is certainly the most fundamental, and two results involving number arithmetic have appeared recently: iterated number multiplication was finally shown to belong to uniform TC^0 [He01, ABH01, CDL] and $(\cup, +, \times)$-circuit evaluation was shown PSPACE-hard [Ya00].

In the boolean setting, the AND and the OR operations combine to capture alternation (in, say, simulations of alternating Turing machines by circuits). In the setting of numbers, perhaps the closest analogs to the AND and the OR become the $\cap$ and the $\cup$. To make sense, this requires an adjustment: since gates will now compute sets of numbers, a $+$-gate and a $\times$-gate having input gates computing $S_1 \subseteq \mathbb{N}$ and $S_2 \subseteq \mathbb{N}$ then compute $\{a+b : a \in S_1, b \in S_2\}$ and $\{a \times b : a \in S_1, b \in S_2\}$ respectively.

Another reason to study such circuits over $\{\cup, \cap, +, \times\}$ is that they obey some form of monotonicity condition: if the set $S \subset \mathbb{N}$ carried by an input gate is replaced by a larger set $S' \supset S$, then the set computed at the output gate of the circuit can only become larger (never smaller). This is reminescent of monotone boolean functions (a boolean function is monotone if flipping an input from 0 to 1 can never change the output from 1 to 0), for which significant complexity bounds are known. How are the circuit and formula evaluation problems over subsets of $\{\cup, \cap, +, \times\}$ related to monotone boolean function complexity?

Here we study the following problem, which we think of as combining number arithmetic with some form of alternation: given a number b and a $\{\cup, \cap, ^-, +, \times\}$-circuit C with number inputs, does b belong to the set computed by the output gate of C? We call this problem $\mathrm{MC}(\cup, \cap, ^-, +, \times)$, and we call $\mathrm{MF}(\cup, \cap, ^-, +, \times)$ the same problem restricted to formulas. Note that a complement gate $^-$ applied to a finite set $S \subset \mathbb{N}$ computes the infinite set $\mathbb{N} \setminus S$. Hence, in the presence of complement gates, the brute force strategy which would exhaustively compute all the sets encountered in the circuit fails. The notation for restricted versions of these problems, for instance $\mathrm{MC}(\cup, +)$, is self-explanatory (see section 2).

Beyond the results stated above, it was known prior to this work that the problem $\mathrm{MF}(\cup, +)$ is NP-complete [SM73], that $\mathrm{MF}(\cup, \cap, ^-, +)$ and $\mathrm{MF}(\cup, ^-, +)$ are PSPACE-complete [SM73], and that $\mathrm{MC}(\cup, +)$ is in PSPACE [Wa84].

Adding results from the present paper, some of whose proofs will be deferred to the full version of the paper for lack of space, we obtain Table I. We highlight here some of the interesting results or techniques:

- The problem $\mathrm{MC}(\cup, \cap, +, \times)$ is NEXPTIME-complete. As an intermediate step, we prove that determining whether a union-intersection-concatenation circuit over a finite alphabet produces a nonempty set is also NEXPTIME-complete.
- The problem $\mathrm{MC}(\cup, \cap, \times)$ is PSPACE-complete.
- The problem $\mathrm{MC}(\cup, \cap, \times)$ reduces in polynomial time to $\mathrm{MC}(\cup, \cap, +)$, which is thus also PSPACE-complete. This reduction is possible because the following problem is solvable in polynomial time: given any set S of numbers excluding 0, compute a set T of pairwise relatively prime numbers and express each $m \in S$ as a product of powers of the numbers in T.
- The problem $\mathrm{MC}(\cup, +)$ is NP-complete. This is a nontrivial improvement over the former PSPACE upper bound.
- The problem $\mathrm{MC}(\cap, +)$ is $\mathrm{C}_=\mathrm{L}$-complete, and so is the problem of testing whether two $\mathrm{MC}(+)$-circuits are equivalent.

The hardest problem we consider is $\mathrm{MC}(\cup, \cap, ^-, +, \times)$. As will be seen, we do not have an upper bound for this problem and it may well be undecidable. On the other hand, Table I shows that its various restrictions hit upon a wealth of complexity classes.

2 Definitions and Known Results

A *circuit* $C = (G, E, g_C)$ is a finite directed acyclic graph (G, E) with a specified node g_C, the *output gate*. The gates with indegree 0 are called the *input gates*.

We consider different types of arithmetic circuits. Let $\mathcal{O} \subseteq \{\cup, \cap, ^-, +, \times\}$. An $\mathcal{O}$*-circuit* $C = (G, E, g_C, \alpha)$ is a circuit (G, E, g_C) whose gates have indegree 0, 1, or 2 and are labelled by the function $\alpha : G \mapsto \mathcal{O} \cup \mathbb{N}$ in the following way: Every input gate g has a label $\alpha(g) \in \mathbb{N}$, every gate g with indegree 1 has the label $\alpha(g) = {}^-$, and every gate g with indegree 2 has a label $\alpha(g) \in \{\cup, \cap, +, \times\}$. For each of its gates g the arithmetic circuit C computes a set $I(g) \subseteq \mathbb{N}$ inductively defined as follows:

- If g is an input gate with label a then $I(g) =_{\text{def}} \{a\}$.
- If g is $+$-gate with pred g_1, g_2 then $I(g) =_{\text{def}} \{k{+}m : k \in I(g_1) \wedge m \in I(g_2)\}$.
- If g is $\times$-gate with pred g_1, g_2 then $I(g) =_{\text{def}} \{k{\cdot}m : k \in I(g_1) \wedge m \in I(g_2)\}$.
- If g is a $\cup$-gate with predecessors g_1, g_2 then $I(g) =_{\text{def}} I(g_1) \cup I(g_2)$.
- If g is a $\cap$-gate with predecessors g_1, g_2 then $I(g) =_{\text{def}} I(g_1) \cap I(g_2)$.
- If g is a ${}^-$-gate with predecessor g_1 then $I(g) =_{\text{def}} \mathbb{N} \setminus I(g_1)$.

The set computed by C is $I(C) =_{\text{def}} I(g_C)$. If $I(g) = \{a\}$ then we also write $I(g) = a$. An $\mathcal{O}$*-formula* is an $\mathcal{O}$-circuit with maximal outdegree 1. For $\mathcal{O} \subseteq \{\cup, \cap, ^-, +, \times\}$ the *membership problems* for $\mathcal{O}$-circuits and $\mathcal{O}$-formulae are

$$\text{MC}(\mathcal{O}) =_{\text{def}} \{(C, b) : C \text{ is an } \mathcal{O}\text{-circuit and } b \in \mathbb{N} \text{ such that } b \in I(C)\}$$

and

$$\text{MF}(\mathcal{O}) =_{\text{def}} \{(F, b) : F \text{ is an } \mathcal{O}\text{-formula and } b \in \mathbb{N} \text{ such that } b \in I(F)\}.$$

For simplicity we write $\text{MC}(o_1, \ldots o_r)$ instead of $\text{MC}(\{o_1, \ldots o_r\})$, and we write $\text{MF}(o_1, \ldots o_r)$ instead of $\text{MF}(\{o_1, \ldots o_r\})$.

Examples. A circuit PRIMES such that $I(\text{PRIMES})$ is the set of prime numbers is obtained by defining the subcircuit GE2 as $\overline{0 \cup 1}$ and defining PRIMES as $\text{GE2} \cap (\overline{\text{GE2} \times \text{GE2}})$. This circuit could easily be turned into a formula. Hence the problem of primality testing easily reduces to $\text{MF}(\cup, \cap, ^-, \times)$. As another example, consider the circuit GOLDBACH defined as $(\text{GE2} \times 2) \cap (\overline{\text{PRIMES} + \text{PRIMES}})$. Then $I(\text{GOLDBACH})$ is empty iff every even number greater than 2 is expressible as a sum of two primes. Hence Goldbach's conjecture holds iff "$0 \in \overline{0 \times \text{GOLDBACH}}$" is a positive instance of $\text{MC}(\cup, \cap, ^-, +, \times)$.

If not otherwise stated, the hardness results in this paper are in terms of many-one logspace reducibility. We assume any circuit and formula encoding in which the gates are sorted topologically and in which immediate predecessors are readily available (say in AC^0). Viewed as graphs, circuits and formulas are not necessarily connected. Numbers are encoded in binary notation.

The following results are known from the literature:

Theorem 1 *1.* [SM73] *The problem* $\text{MF}(\cup, +)$ *is* NP*-complete.*
2. [SM73] *The problems* $\text{MF}(\cup, \cap, ^-, +)$ *and* $\text{MF}(\cup, ^-, +)$ *are* PSPACE*-complete.*
3. [Wa84] *The problem* $\text{MC}(\cup, +)$ *is in* PSPACE.
4. [Ya00] *The problem* $\text{MC}(\cup, +, \times)$ *is* PSPACE*-complete.*

In [Wa84] it is shown that the problem $\mathrm{MC}(\cup,+)$ restricted to $(\cup,+)$-circuits for which every $+$-gate has at least one input gate as predecessor is NP-complete.

By De Morgan's laws we have

Proposition 2 *For every $\mathcal{O} \subseteq \{+,\times\}$,*

1. $\mathrm{MC}(\{\cup,\cap,^-\} \cup \mathcal{O}) \equiv_{\mathrm{m}}^{\mathrm{p}} \mathrm{MC}(\{\cup,^-\} \cup \mathcal{O}) \equiv_{\mathrm{m}}^{\mathrm{p}} \mathrm{MC}(\{\cap,^-\} \cup \mathcal{O})$.
2. $\mathrm{MF}(\{\cup,\cap,^-\} \cup \mathcal{O}) \equiv_{\mathrm{m}}^{\mathrm{p}} \mathrm{MF}(\{\cup,^-\} \cup \mathcal{O}) \equiv_{\mathrm{m}}^{\mathrm{p}} \mathrm{MF}(\{\cap,^-\} \cup \mathcal{O})$.

Hence we can omit $\mathrm{MC}(\{\cup,^-\} \cup \mathcal{O})$, $\mathrm{MC}(\{\cap,^-\} \cup \mathcal{O})$, $\mathrm{MF}(\{\cup,^-\} \cup \mathcal{O})$, and $\mathrm{MF}(\{\cap,^-\} \cup \mathcal{O})$ from our exhaustive study.

3 Multiplication versus Addition

In this section we will establish a relationship between the complexity of the membership problems for $(\mathcal{O} \cup \{\times\})$-circuits and $(\mathcal{O} \cup \{+\})$-circuits, for $\mathcal{O} \subseteq \{\cup,\cap,^-\}$. To this end we need the following problem. Let $\gcd(a,b)$ be the greatest common divisor of the numbers $a,b \geq 1$.

Gcd-Free Basis (GFB)
Given: Numbers $a_1, a_2, \ldots, a_n \geq 1$.
Compute: Numbers $m \geq 1$, $q_1, \ldots, q_m \geq 2$, and $e_{11}, \ldots, e_{nm} \geq 0$ such that $\gcd(q_i, q_j) = 1$ for $i \neq j$ and $a_i = \prod_{j=1}^{m} q_j^{e_{ij}}$ for $i = 1, \ldots, n$.

Despite the fact that factoring may not be possible in polynomial time, the following is known (see [BS96]):

Proposition 3 *Gcd-Free Basis can be computed in polynomial time.*

As an auxiliary tool we need the generalized membership problems $\mathrm{MC}^*(\mathcal{O})$ and $\mathrm{MF}^*(\mathcal{O})$ for arithmetic circuit and formulae, resp., with addition. These problems deal with elements of $\mathbb{N}^m \cup \{\infty\}$, for an $m \geq 1$ prescribed on input, where the addition on m-tuples is defined componentwise and $a + \infty = \infty + a = \infty + \infty = \infty$ for every $a \in \mathbb{N}^m$. Note that polynomial space many-one reducibility is understood to be performed by polynomial space computable polynomially bounded functions.

Lemma 4 *1. For $\mathcal{O} \subseteq \{\cup,\cap\}$, the problem $\mathrm{MC}(\mathcal{O} \cup \{\times\})$ is polynomial time many-one reducible to the problem $\mathrm{MC}^*(\mathcal{O} \cup \{+\})$.*
2. For $\mathcal{O} \subseteq \{\cup,\cap\}$, the problem $\mathrm{MF}(\mathcal{O} \cup \{\times\})$ is polynomial time many-one reducible to the problem $\mathrm{MF}^*(\mathcal{O} \cup \{+\})$.
3. For $\mathcal{O} \subseteq \{\cup,\cap,^-\}$, the problem $\mathrm{MC}(\mathcal{O} \cup \{\times\})$ is polynomial space many-one reducible to the problem $\mathrm{MC}^*(\mathcal{O} \cup \{+\})$.
4. For $\mathcal{O} \subseteq \{\cup,\cap,^-\}$, the problem $\mathrm{MF}(\mathcal{O} \cup \{\times\})$ is polynomial space many-one reducible to the problem $\mathrm{MF}^*(\mathcal{O} \cup \{+\})$.

Proof. 1. Let $\mathcal{O} \subseteq \{\cup,\cap\}$, let C be a $(\mathcal{O} \cup \{\times\})$-circuit with the input gates $u_1, \ldots, u_s$, and let $b \in \mathbb{N}$. Observe that the absence of $+$ in C entails that any number in $I(C)$ is expressible as a monomial in the inputs. Compute in polynomial time (Lemma 3) numbers $q_1, \dots, q_m \geq 2$ and $e_{11}, \dots, e_{sm}, e_1, \dots, e_m \geq 1$ such that $\gcd(q_i, q_j) = 1$ for $i \neq j$, $\alpha(u_i) = \prod_{j=1}^m q_j^{e_{ij}}$ for $i = 1, \dots, s$ such that $\alpha(u_i) > 0$ and $b = \prod_{j=1}^m q_j^{e_j}$ if $b > 0$. Let $M =_{\rm def} \{ \prod_{j=1}^m q_j^{f_j} : f_1, \dots, f_m \geq 0 \} \subseteq \mathbb{N}$ and let $\sigma : M \cup \{0\} \to \mathbb{N}^m \cup \{\infty\}$ be defined by $\sigma(\prod_{j=1}^m q_j^{f_j}) =_{\rm def} (f_1, \dots, f_m)$ and $\sigma(0) = \infty$. Obviously, σ is a monoid isomorphism between $(M \cup \{0\}, \times)$ and $(\mathbb{N}^m \cup \{\infty\}, +)$ where we define $\infty + \infty = \infty + a = a + \infty = \infty$ for every $a \in \mathbb{N}^m$. Because $M \cup \{0\}$ is closed under $\times$, the set of numbers computed by any gate in C is included in $M \cup \{0\}$. Furthermore, the following holds for any $\emptyset \subseteq S_1, S_2 \subseteq M \cup \{0\}$:

$$\begin{aligned} \sigma(S_1 \times S_2) &= \sigma(S_1) + \sigma(S_2), \\ \sigma(S_1 \cup S_2) &= \sigma(S_1) \cup \sigma(S_2), \\ \sigma(S_1 \cap S_2) &= \sigma(S_1) \cap \sigma(S_2). \end{aligned}$$

The reduction therefore consists of converting C into a $(\mathcal{O} \cup \{+\})$-circuit C' which has the same structure as C where a $\times$-gate in C becomes a $+$-gate in C' and an input gate u_i gets label $\sigma(\alpha(u_i))$. An induction using the three identities above shows that for all $a \in M \cup \{0\}$ the following holds: $a \in I(v)$ in $C \Leftrightarrow \sigma(a) \in I(v)$ in C'. This concludes the proof because $b \in M \cup \{0\}$.

2. Same as above because the construction preserves the circuit structure.

3. and 4. If we have complementation then it is no longer true that every number computed within C has a decomposition into $q_1, \dots, q_m$. To salvage the above construction, the isomorphism σ must therefore be extended to convey information about $\mathbb{N} \setminus M$. A slick way to do this is to begin from the *full* prime decomposition of the numbers $u_1, \dots, u_s, b$ and to trade the former isomorphism for a homomorphism. Indeed let $q_1, \dots, q_m$ exhaust the distinct prime divisors of $u_1, \dots, u_s, b$, and let $q_1, \dots, q_m, q_{m+1}, q_{m+2}, \dots$ be the sequence of all primes (in some order). For every number $\prod_{j=1}^\infty q_j^{d_j} \in \mathbb{N} \setminus \{0\}$ define $\sigma(\prod_{j=1}^\infty q_j^{d_j}) =_{\rm def} (d_1, \dots, d_m, \sum_{j>m} d_j)$, and define $\sigma(0) =_{\rm def} \infty$. Let $M =_{\rm def} \{ \prod_{j=1}^m q_j^{f_j} : f_1, \dots, f_m \geq 0 \} \subseteq \mathbb{N}$. Because the full prime decomposition was used, σ is a well-defined monoid homomorphism from $(\mathbb{N}, \times)$ onto $(\mathbb{N}^{m+1} \cup \{\infty\}, +)$, where $\sigma(M \cup \{0\}) = (\mathbb{N}^m \otimes \{0\}) \cup \{\infty\}$ (σ is one-one on this part; $\otimes$ denotes direct product) and $\sigma(\mathbb{N} \setminus (M \cup \{0\})) = (\mathbb{N}^m \otimes (\mathbb{N} \setminus \{0\}))$.

The reduction then again consists of converting C into a $(\mathcal{O} \cup \{+\})$-circuit C' having the same structure as C where a $\times$-gate in C becomes a $+$-gate in C' and an input gate u_i gets label $\sigma(\alpha(u_i))$. An induction proves that for any $a \in \mathbb{N}$ and any gate v in C, $a \in I(v)$ in $C \Leftrightarrow \sigma(a) \in I(v)$ in C'. This implies that $b \in I(C) \Leftrightarrow \sigma(b) \in I(C')$, completing the proof. The polynomial space is needed to perform the prime decomposition (if needed, a possibly weaker reducibility, like a many-one polynomial time reduction with an NP oracle, would suffice).

□

In some cases the generalized membership problems used above are logspace equivalent to their standard versions:

Lemma 5 *Let* $\{\cup\} \subseteq \mathcal{O} \subseteq \{\cup, \cap\}$.

1. $\mathrm{MC}^*(\mathcal{O} \cup \{+\}) \equiv_{\mathrm{m}}^{\log} \mathrm{MC}(\mathcal{O} \cup \{+\})$.
2. $\mathrm{MF}^*(\mathcal{O} \cup \{+\}) \equiv_{\mathrm{m}}^{\log} \mathrm{MF}(\mathcal{O} \cup \{+\})$.

4 NP-Complete Membership Problems

Lemma 6 *The problem* $\mathrm{MF}(\cup, \cap, +, \times)$ *is in* NP.

Proof sketch. An NP-algorithm can guess a proof that $b \in I(F)$ and can check that the input gates used in the proof carry the required values. □

The following is a nontrivial improvement over the known PSPACE upper bound for $\mathrm{MC}(\cup, +)$:

Lemma 7 *The problems* $\mathrm{MC}(\cup, +)$ *and* $\mathrm{MC}(\cup, \times)$ *are in* NP.

Proof. Let C be a $\{\cup, +\}$-circuit, and let T_C be the tree which is the result of unfolding C into a tree. A subtree T of T_C is called *computation tree* of C iff

- the output gate of T_C is in T,
- both predecessors of a $+$-gate of T are in T, and
- exactly one predecessor of a $\cup$-gate of T is in T.

Hence T describes one of the many ways to compute a number from $I(C)$.

A gate g in C corresponds to several copies of it in T_C (and hence also in a computation tree T of C). Let $\beta_{C,T}(g)$ be the number of copies of g in T. In the same way, an edge in C corresponds to several copies of it in T_C (and hence also in a computation tree T of C). Let $\beta_{C,T}(e)$ be the number of copies of e in T.

Defining $\mathrm{s}(C,T) =_{\mathrm{def}} \sum_{\text{input gate } g \text{ of } C} \alpha(g) \cdot \beta_{C,T}(g)$

we obtain immediately $I(C) = \{\mathrm{s}(C,T) : T \text{ is a computation tree of } C\}$.

A function $\beta : G \cup E \mapsto \mathbb{N}$ is a *valuation function* of the $\{\cup, +\}$-circuit $C = (G, E, \alpha)$ if the following holds:

- $\beta(g_C) = 1$,
- if g is a $+$-gate with the incoming edges e_1 and e_2 then $\beta(g) = \beta(e_1) = \beta(e_2)$,
- if g is a $\cup$-gate with the incoming edges e_1 and e_2 then $\beta(g) = \beta(e_1) + \beta(e_2)$,
- if g is a gate with the outgoing edges $e_1, \ldots e_k$ then $\beta(g) = \beta(e_1) + \ldots + \beta(e_k)$.

For a valuation function β of C define $\mathrm{s}(C,\beta) =_{\mathrm{def}} \sum_{\text{input gate } g \text{ of } C} \alpha(g) \cdot \beta(g)$.

See the full paper for the proof of the following two claims:

Claim 1: If T is a computation tree of C then there exists a valuation function β of C such that $\mathrm{s}(C,\beta) = \mathrm{s}(C,T)$.

Claim 2: If β is a valuation function of C then there exists a computation tree T of C such that $\mathrm{s}(C,T) = \mathrm{s}(C,\beta)$.

Then we obtain $I(C) = \{\mathrm{s}(C,\beta) : \beta \text{ is a valuation function of } C\}$, and hence $a \in I(C) \Leftrightarrow \exists \beta(\beta$ is a valuation function of C and $\mathrm{s}(C,\beta) = a)$. However, the latter property is in NP. This completes the proof of $\mathrm{MC}(\cup, +) \in \mathrm{NP}$. From this, Lemma 4, and Lemma 5 we obtain $\mathrm{MC}(\cup, \times) \in \mathrm{NP}$. □

Lemma 8 $\mathrm{MF}(\cup, +)$ *and* $\mathrm{MF}(\cup, \times)$ *are* NP*-hard.*

Proof. The NP-hardness of $\mathrm{MF}(\cup,+)$ is known ([SM73], cf. Theorem 1). We prove in the full paper that 3-SAT $\leq_{\mathrm{m}}^{\log} \mathrm{MF}(\cup,\times)$. $\square$

As an immediate consequence of the preceding lemmas we obtain:

Theorem 9 *The problems* $\mathrm{MF}(\cup,\cap,+,\times)$, $\mathrm{MF}(\cup,\cap,+)$, $\mathrm{MF}(\cup,\cap,\times)$, $\mathrm{MF}(\cup,+,\times)$, $\mathrm{MF}(\cup,+)$, $\mathrm{MF}(\cup,\times)$, $\mathrm{MC}(\cup,+)$, *and* $\mathrm{MC}(\cup,\times)$ *are* NP*-complete.*

5 PSPACE-Complete Membership Problems

Theorem 10 *1.* [Ya00] *The problem* $\mathrm{MC}(\cup,+,\times)$ *is* PSPACE*-complete.*
2. [SM73] *The problem* $\mathrm{MF}(\cup,\cap,^-,+)$ *is* PSPACE*-complete.*

Lemma 11 *The problems* $\mathrm{MF}(\cup,\cap,^-,\times)$, $\mathrm{MC}(\cup,\cap,\times)$, *and* $\mathrm{MC}(\cup,\cap,+)$ *are* PSPACE*-hard, the latter w.r.t. polytime reducibility.*

Proof. A single proof in the full paper shows that the quantified boolean 3-CNF formula problem, which is known to be PSPACE-complete [SM73], is logspace many-one reducible to the problems $\mathrm{MF}(\cup,\cap,^-,\times)$ and $\mathrm{MC}(\cup,\cap,\times)$. The hardness of $\mathrm{MC}(\cup,\cap,+)$ then follows by Lemma 4 and Lemma 5. $\square$

Lemma 12 *For every gate g of a generalized $(\cup,\cap,^-,+)$-circuit, if $I(g) \neq \emptyset$ then $I(g) \cap (\{0,1,\ldots,2^{|C|+1}\}^m \cup \{\infty\}) \neq \emptyset$.*

Lemma 13 *The problems* $\mathrm{MC}(\cup,\cap,^-,+)$ *and* $\mathrm{MC}(\cup,\cap,^-,\times)$ *are in* PSPACE*.*

Proof. Note that $\mathrm{MC}(\cup,\cap,^-,+)$ reduces to $\mathrm{MC}^*(\cup,\cap,^-,+)$ simply by replacing every negation gate $\overline{g}$ with $\overline{g} \cap \overline{\infty}$. To prove Lemma 13, it then suffices by Lemma 4 to show that $\mathrm{MC}^*(\cup,\cap,^-,+) \in \mathrm{PSPACE}$.

For a generalized $\{\cup,\cap,^-,+\}$-circuit C and $b \in \mathbb{N}^m \cup \{\infty\}$, the idea is to use alternating polynomial time to guess an (alternating) proof that $b \in I(C)$. A subtlety arises when a $+$-gate g is encountered and it is guessed that ∞ is fed into g by one of its two inputs. Then we seek a witness to the fact that the other input to g carries a nonempty set. Now how large can such a witness be? Lemma 12 ensures that if a witness exists at all, then a witness exists of polynomial length. The details are given in the full paper.

As a direct consequence of the preceding lemmas we obtain:

Theorem 14 *The five problems* $\mathrm{MC}(\cup,\cap,^-,+)$, $\mathrm{MC}(\cup,\cap,+)$, $\mathrm{MC}(\cup,\cap,^-,\times)$, $\mathrm{MC}(\cup,\cap,\times)$, *and* $\mathrm{MF}(\cup,\cap,^-,\times)$ *are* PSPACE*-complete, the latter w.r.t. polytime reducibility. The problem* $\mathrm{MF}(\cup,\cap,^-,+,\times)$ *is* PSPACE*-hard.*

6 Beyond PSPACE

As an auxiliary tool we introduce the following $(\cup,\cap,\cdot)$-circuits which compute finite sets of words. Such a circuit C only has gates of indegrees 0 and 2. Every input gate (i.e., gate of indegree 0) g is labelled with a word from a given alphabet

Σ^*, and every gate of in degree 2 is labelled with $\cup$, $\cap$, or $\cdot$ (concatenation). For each of its gates g, the circuit computes a set $I(g) \subseteq \Sigma^*$ inductively defined as follows: If g is an input gate with label v then $I(g) =_{\rm def} \{v\}$. If g is an ω-gate ($\omega \in \{\cup,\cap,\cdot\}$) with predecessors g_l, g_r then $I(g) =_{\rm def} I(g_l)\,\omega\, I(g_r)$. Finally, $I(C) =_{\rm def} I(g_C)$ where g_C is the output gate of C. A $(\cup,\cap,\cdot)$-circuit C is called *special* if for every gate g of C there exists a $k \geq 0$ such that $I(g) \subseteq \Sigma^k$. Let $\mathrm{NE}(\cup,\cap,\cdot)$ be the nonemptyness problem for special $(\cup,\cap,\cdot)$-circuits, i.e., $\mathrm{NE}(\cup,\cap,\cdot) =_{\rm def} \{C : C \text{ is a special } (\cup,\cap,\cdot)\text{-circuit such that } I(C) \neq \emptyset\}$.

Lemma 15 *The problem* $\mathrm{NE}(\cup,\cap,\cdot)$ *is* NEXPTIME-*hard.*

Proof. A delicate generic reduction is given in the full paper. Care is needed to recursively construct sets of words capable of ensuring the match between equal length subwords representing successive machine configurations. □

Theorem 16 *The problem* $\mathrm{MC}(\cup,\cap,+,\times)$ *is* NEXPTIME-*complete.*

Proof. 1. To prove $\mathrm{MC}(\cup,\cap,+,\times)$ is NEXPTIME-hard we show $\mathrm{NE}(\cup,\cap,\cdot) \leq_{\rm m}^{\log} \mathrm{MC}(\cup,\cap,+,\times)$. For $w \in \{0,1\}^*$ let $\mathrm{bin}^{-1}(w)$ be that natural number whose binary description is w (possibly with leading zeros), and for $L \subseteq \{0,1\}^*$ let $\mathrm{bin}^{-1}(L) =_{\rm def} \{\mathrm{bin}^{-1}(w) : w \in L\}$.

Given a special $(\cup,\cap,\cdot)$-circuit C such that $I(C) \subseteq \{0,1\}^k$ (using a block encoding this can be assumed without loss of generality) we construct in logarithmic space a $(\cup,\cap,+,\times)$-circuit C' such that $I(C') = \mathrm{bin}^{-1}(I(C))$. The circuit C' basically has the same structure as C: An input gate in C with label w becomes an input gate in C' with label $\mathrm{bin}^{-1}(w)$, a $\cup$-gate in C becomes a $\cup$-gate in C', and a $\cap$-gate in C becomes a $\cap$-gate in C'. A $\cdot$-gate g in C with predecessor g_1, g_2 such that $I(g_2) \subseteq \{0,1\}^k$ is replaced in C' by a subcircuit which computes $\mathrm{bin}^{-1}(I(g_1)\cdot I(g_2)) = (2^k \times \mathrm{bin}^{-1}(I(g_1))) + \mathrm{bin}^{-1}(I(g_2))$. (Here it is important that C is a *special* $(\cup,\cap,\cdot)$-circuit.)

Now, $I(C) \neq \emptyset \Leftrightarrow I(C') \neq \emptyset \Leftrightarrow 0 \in (\{0\} \times I(C'))$.

2. To see that $\mathrm{MC}(\cup,\cap,+,\times)$ is in NEXPTIME, simply unfold a given $(\cup,\cap,+,\times)$-circuit into a (possibly exponentially larger) $(\cup,\cap,+,\times)$-formula and apply the NP-algorithm from Lemma 6. □

Corollary 17 *The problem* $\mathrm{NE}(\cup,\cap,\cdot)$ *is* NEXPTIME-*complete.*

As an immediate consequence of Theorem 16 we obtain also:

Theorem 18 *The problem* $\mathrm{MC}(\cup,\cap,^-,+,\times)$ *is* NEXPTIME-*hard.*

Remark. Since $\mathrm{MC}(\cup,\cap,^-,+,\times) \equiv_{\rm m}^{\log} \overline{\mathrm{MC}(\cup,\cap,^-,+,\times)}$ these problems cannot be in NEXPTIME unless NEXPTIME = co-NEXPTIME. In fact, there is evidence suggesting that $\mathrm{MF}(\cup,\cap,^-,+,\times)$ might not be decidable. Indeed, Christian Glaßer in Würzburg was the first to observe that there is a simple $\{\cup,\cap,^-,+,\times\}$-formula G (see the examples given in Section 2) having the property that $(G,0) \in \mathrm{MF}(\cup,\cap,^-,+,\times)$ if and only if Goldbach's Conjecture is true. Hence, a decision procedure for $\mathrm{MF}(\cup,\cap,^-,+,\times)$ would provide a terminating algorithm to test Goldbach's Conjecture; this would be surprising.

7 P-Complete Membership Problems

Theorem 19 *The problem* MC$(+,\times)$ *is* P*-complete.*

Theorem 20 *The problems* MC$(\cup,\cap)$ *and* MC$(\cup,\cap,^{-})$ *are* P*-complete.*

Proof. To prove MC$(\cup,\cap,^{-}) \in$ P, let C be a $(\cup,\cap,^{-})$-circuit and $b \in \mathbb{N}$. Define $S =_{\text{def}} \bigcup_{v \text{ input gate}} I(v)$. We prove that (*) for every gate v there are sets $P, N \subseteq S$ such that $I(v) = P \cup \overline{N}$. Then we can compute in polynomial time all $I(v)$ from the inputs down to the output by storing only the sets P and N.

To see (*) let $P_1, P_2, N_1, N_2 \subseteq S$ and observe $(P_1 \cup \overline{N_1}) \cup (P_2 \cup \overline{N_2}) = (P_1 \cup P_2) \cup \overline{(N_1 \cap N_2)}$, $(P_1 \cup \overline{N_1}) \cap (P_2 \cup \overline{N_2}) = ((P_1 \cap P_2) \cup (P_1 \setminus N_2) \cup (P_2 \setminus N_1)) \cup \overline{(N_1 \cup N_2)}$, and $\overline{P_1 \cup \overline{N_1}} = N_1 \setminus P_1$.

The hardness proof is by showing that the P-complete *monotone boolean circuit value problem* can be reduced to MC$(\cup,\cap)$. To do so we convert a monotone boolean circuit C into a $(\cup,\cap)$-circuit C' of almost the same structure where every input gate in C with boolean value 1 becomes an input gate in C' with integer value 1, every input gate in C with boolean value 0 becomes an $\cap$-gate in C' with two input gates with labels 0 and 1, resp., as predecessors, every $\vee$-gate in C becomes a $\cup$-gate in C', and every $\wedge$-gate in C becomes a $\cap$-gate in C'. It is easy to see that v evaluates to 0 in C if and only if $I(v) = \emptyset$ in C', and v evaluates to 1 in C if and only if $I(v) = \{1\}$ in C'. Hence, C evaluates to 1 if and only if $1 \in I(C')$. □

8 Circuits with Intersection as the Only Set Operation

Circuits with intersection as the only set operation are special in the sense that every node computes a singleton or the empty set. Thus these circuits bear some relationship to circuits of the same type without intersection. For $\mathcal{O} \subseteq \{+,\times\}$ define EQ$(\mathcal{O}) =_{\text{def}} \{(C_1, C_2) : C_1, C_2$ are $\mathcal{O}$-circuits such that $I(C_1) = I(C_2)\}$.

Lemma 21 *1.* MC$(\cap,+) \leq_{\text{m}}^{\log}$ EQ$(+)$
2. MC$(\cap,+,\times) \equiv_{\text{m}}^{\log}$ EQ$(+,\times)$

Proof. Note that an empty set computed at any accessible gate of a $\{\cap,+,\times\}$-circuit propagates to the output. The reductions from left to right consist of progressively bypassing $\cap$-gates, creating for each such gate g a pair (C_g, C'_g) of $\cap$-free subcircuits corresponding to the inputs to g, and rigging in the end two $\cap$-free circuits that are equivalent iff C_g is equivalent to C'_g in all the pairs created from accessible gates. See the full paper for details. □

For the following we need the complexity classes #L and $\text{C}_=\text{L}$. For a nondeterministic logarithmic space machine M, define $\text{n}_M(x)$ as the number of accepting paths of M on input x. The class #L precisely consists of these functions n_M. A set A is in $\text{C}_=\text{L}$ if and only if there exist $f \in$ #L and a logarithmic space computable function g such that $x \in A \Leftrightarrow f(x) = g(x)$ for every x. For a survey on these and other counting classes see [All97].

Observe that this definition is equivalent to: A set A is in $\mathrm{C_{=}L}$ if and only if there exist $f, g \in \#\mathrm{L}$ such that $x \in A \Leftrightarrow f(x) = g(x)$ for every x.

Theorem 22 *$\mathrm{MC}(\cap,+)$, $\mathrm{MC}(+)$, and $\mathrm{EQ}(+)$ are $\leq_{\mathrm{m}}^{\log}$-complete for $\mathrm{C_{=}L}$.*

Theorem 23 *1. The problem $\mathrm{MC}(\cap,+,\times)$ is in co-NP.* [1]
2. The problem $\mathrm{MC}(\cap,+,\times)$ is P-hard.

Theorem 24 *1. The problem $\mathrm{MC}(\cap,\times)$ is in P.*
2. The problem $\mathrm{MC}(\cap,\times)$ is $\mathrm{C_{=}L}$-hard.

Theorem 25 *The problem $\mathrm{MC}(\times)$ is NL-complete.*

Proof. Hardness is argued in the full paper. Here we describe a many-one reduction from $\mathrm{MC}(\times)$ to the iterated multiplication decision problem $\mathrm{IMD} =_{\mathrm{def}} \{(a_1,\ldots a_r, b) : a_1,\ldots a_r, b \in \mathbb{N} \wedge \prod_{i=1}^{r} a_i = b\}$ via a function which is logspace computable with an oracle from NL. Since $\mathrm{IMD} \in \mathrm{L}$ ([CDL], or better yet, uniform TC^0 [ABH01, He01]) we obtain $\mathrm{MC}(\times) \in \mathrm{L}^{\mathrm{NL}}$. However, $\mathrm{L}^{\mathrm{NL}} = \mathrm{NL}$.

Let C be a $(\times)$-circuit and $b \in \mathbb{N}$. Let $g_1, \ldots g_r$ be the input gates of C, let g_C be the output gate of C, and let $n(C, g, g')$ be the number of different paths in C from gate g to gate g'. Obviously, $n \in \#\mathrm{L}$, and we obtain $I(C) = \prod_{i=1}^{r} I(g_i)^{n(C,g_i,g_C)}$. Defining $s(i) =_{\mathrm{def}} \min\{n(C, g_i, g_C), |b|+1\}$ for $i = 1, \ldots, r$ we obtain

$$\begin{aligned} b \in I(C) &\Leftrightarrow b = \textstyle\prod_{i=1}^{r} I(g_i)^{n(C,g_i,g_C)} \Leftrightarrow b = \prod_{i=1}^{r} I(g_i)^{s(i)} \\ &\Leftrightarrow (\underbrace{I(g_1),\ldots,I(g_1)}_{s(1)},\ldots,\underbrace{I(g_r),\ldots,I(g_r)}_{s(r)}, b) \in \mathrm{IMD} \end{aligned}$$

The latter tuple of numbers is generated as follows: for $i = 1, \ldots r$ and $k = 1, \ldots, |b|+1$ ask $n(C, g_i, g_C) \geq k$, which are queries to a NL-set [ARZ99]. (This owes to the fact that only small k, i.e. k whose values are polynomially bounded in the length of the input, are considered.) If such a query $n(C, g_i, g_C) \geq k$ is answered in the affirmative, output $I(g_i)$, and finally output b. □

9 Further Results

The following results are argued in the full paper:

Theorem 26 *The problems $\mathrm{MC}(\cup)$ and $\mathrm{MC}(\cap)$ are NL-complete.*

Theorem 27 *The problems $\mathrm{MF}(\cap,+,\times)$ and $\mathrm{MF}(+,\times)$ are in DLOGCFL.*

Theorem 28 *The problems $\mathrm{MF}(\cup,\cap,^-)$, $\mathrm{MF}(\cup,\cap)$, $\mathrm{MF}(\cup)$, $\mathrm{MF}(\cap)$, $\mathrm{MF}(\cap,+)$, $\mathrm{MF}(\cap,\times)$, $\mathrm{MF}(+)$ and $\mathrm{MF}(\times)$ are L-complete under AC^0-reducibility.*

The L-hardness of the MF problems considered in this section owe to our choice of formula encoding. At such low complexity levels, a more appropriate choice is infix notation; then $\mathrm{MF}(\cup,\cap,^-)$ becomes NC^1-complete by equivalence with the Boolean formula value problem [Bu87], and some of the other restrictions considered in Theorem 28 drop down into yet smaller classes.

[1] Christian Glaßer, Würzburg, recently proved that $\mathrm{MC}(\cap,+,\times)$ is in co-R.

10 Conclusion

Table I summarizes the known complexity status of the membership problems for arithmetic circuits over subsets of $\mathbb{N}$. Several open questions are apparent from the table, most notably that of finding an upper bound (if one exists) on the complexity of $\mathrm{MC}(\cup, \cap, ^-, +, \times)$.

We observe that the problems $\mathrm{MC}(\times)$ and $\mathrm{MC}(+)$, complete for NL and for the $\mathrm{C_=L}$ respectively, offer an interesting new perspective on these two classes. If one could reduce $\mathrm{MC}(+)$ to $\mathrm{MC}(\times)$, then it would follow that $\mathrm{NL} = \mathrm{C_=L}$.

Acknowledgements. We are grateful to Eric Allender helping us with the upper bound in Theorem 25, to Heribert Vollmer and Christian Glaßer for very useful discussions, and to the anonymous referees for valuable suggestions, including the need to correct our choice of formula encoding and to clarify its ramifications.

$\mathcal{O}$	MC($\mathcal{O}$) lower bound	MC($\mathcal{O}$) upper bound	Th.	MF($\mathcal{O}$) low. bound	MF($\mathcal{O}$) upp. bound	Th.
$\cup, \cap, ^-, +, \times$	NEXPTIME	?	18	PSPACE	?	14
$\cup, \cap, +, \times$	NEXPTIME	NEXPTIME	16	NP	NP	9
$\cup, +, \times$	PSPACE	PSPACE	10	NP	NP	9
$\cap, +, \times$	P	co-R	23	L	DLOGCFL	27
$+, \times$	P	P	19	L	DLOGCFL	27
$\cup, \cap, ^-, +$	PSPACE	PSPACE	14	PSPACE	PSPACE	10
$\cup, \cap, +$	PSPACE	PSPACE	14	NP	NP	9
$\cup, +$	NP	NP	9	NP	NP	9
$\cap, +$	$\mathrm{C_=L}$	$\mathrm{C_=L}$	22	L	L	28
$+$	$\mathrm{C_=L}$	$\mathrm{C_=L}$	22	L	L	28
$\cup, \cap, ^-, \times$	PSPACE	PSPACE	14	PSPACE	PSPACE	14
$\cup, \cap, \times$	PSPACE	PSPACE	14	NP	NP	9
$\cup, \times$	NP	NP	9	NP	NP	9
$\cap, \times$	$\mathrm{C_=L}$	P	24	L	L	28
$\times$	NL	NL	25	L	L	28
$\cup, \cap, ^-$	P	P	20	L	L	28
$\cup, \cap$	P	P	20	L	L	28
$\cup$	NL	NL	26	L	L	28
$\cap$	NL	NL	26	L	L	28

TABLE I: State of the art. The results on $\mathrm{MF}(\cup, +)$, $\mathrm{MF}(\cup, \cap, ^-, +)$ as well as on $\mathrm{MC}(\cup, +, \times)$ were already known from the literature, please refer to the relevant sections for the appropriate credit. Lower bounds of course refer to hardness results.

References

[AAD00] M. Agrawal, E. Allender, and S. Datta, On TC^0, AC^0, and arithmetic circuits, *J. Computer and System Sciences* 60 (2000), pp. 395–421.

[All97] E. Allender, Making computation count: Arithmetic circuits in the Nineties, in the Complexity Theory Column, *SIGACT NEWS* 28 (4) (1997) pp. 2-15.

[ARZ99] E. Allender, K. Reinhardt, S. Zhou, Isolation, matching, and counting: Uniform and nonuniform upper bounds, *J. Computer and System Sciences* 59(1999), pp. 164–181.

[ABH01] E. Allender, D. Barrington, and W. Hesse, Uniform constant-depth threshold circuits for division and iterated multiplication, *Proceedings 16th Conference on Computational Complexity*, 2001, pp. 150–159.

[AJMV98] E. Allender, J. Jiao, M. Mahajan and V. Vinay, Non-commutative arithmetic circuits: depth-reduction and depth lower bounds, *Theoretical Computer Science* Vol. 209 (1,2) (1998), pp. 47-86.

[BS96] E. Bach, J. Shallit, Algorithmic Number Theory, Volume I: Efficient Algorithms, MIT Press 1996.

[BM95] M. Beaudry and P. McKenzie, Circuits, matrices and nonassociative computation, *J. Computer and System Sciences* 50 (1995), pp. 441–455.

[BMPT97] M. Beaudry, P. McKenzie, P. Péladeau, D. Thérien, Finite monoids: from word to circuit evaluation, *SIAM J. Computing* 26 (1997), pp. 138–152.

[BCGR92] S. Buss, S. Cook, A. Gupta, V. Ramachandran, An optimal parallel algorithm for formula evaluation, *SIAM J. Computing* 21 (1992), pp. 755–780.

[Bu87] S. R. Buss, The boolean formula value problemis in ALOGTIME, Proceedings 19th ACM Symp, on the Theory of Computing, 1987, pp. 123–131.

[CMTV98] H. Caussinus, P. McKenzie, D. Thérien, H. Vollmer, Nondeterministic NC^1 computation, *J. Computer and System Sciences*, 57 (2), 1998, pp. 200–212.

[CSV84] A. K. Chandra, L. Stockmeyer, U. Vishkin, Constant depth reducibility, *SIAM Journal on Computing*, 13, 1984, pp. 423–439.

[CDL] A. Chiu, G. Davida, and B. Litow, *NC^1 division*, available at http://www.cs.jcu.edu.au/ bruce/papers/cr00_3.ps.gz

[Go77] L. M. Goldschlager, The monotone and planar circuit value problems are logspace complete for P, *SIGACT News*, 9, 1977, pp. 25–29.

[He01] W. Hesse, Division in uniform TC^0, Proceedings of the 28th International Colloquium on Automata, Languages, and Programming 2001, Lecture Notes in Computer Science 2076, pp. 104–114

[Ga84] J. von zur Gathen, Parallel algorithms for algebraic problems, *SIAM J. on Computing* 13(4), (1984), pp. 802-824.

[GHR95] R. Greenlaw, J. Hoover and L. Ruzzo, *Limits to parallel computation, P-completeness theory,* Oxford University Press, 1995, 311 pages.

[Og98] M. Ogihara, The PL hierarchy collapses, *SIAM Journal of Computing*, 27, 1998, pp. 1430–1437.

[SM73] L. J. Stockmeyer, A. R. Meyer, Word Problems Requiring Exponential Time, Proceedings 5th ACM Symposium on the Theory of Computing, 1973, pp. 1–9.

[Vo00] H. Vollmer, Circuit complexity, Springer, 2000.

[Wa84] K. W. Wagner, The complexity of problems concerning graphs with regularities, Proceedings 11th Mathematical Foundations of Computer Science 1984, Lecture Notes in Computer Science 176, pp. 544–552. Full version as TR N/84/52, Friedrich-Schiller-Universität Jena, 1984.

[Ya00] K. Yang, Integer circuit evaluation is PSPACE-complete, Proceedings 15th Conference on Computational Complexity, 2000, pp. 204–211.