Electronic Colloquium on Computational Complexity, Report No. 197 (2010)

# Mean-payoff games and propositional proofs*

Albert Atserias †
Universitat Politècnica de Catalunya
Barcelona, Spain

Elitza Maneva‡
Universitat de Barcelona
Barcelona, Spain

May 25, 2010

**Abstract**

We associate a CNF-formula to every instance of the mean-payoff game problem in such a way that if the value of the game is non-negative the formula is satisfiable, and if the value of the game is negative the formula has a polynomial-size refutation in $\Sigma_2$-Frege (i.e. DNF-resolution). This reduces mean-payoff games to the weak automatizability of $\Sigma_2$-Frege, and to the interpolation problem for $\Sigma_{2,2}$-Frege. Since the interpolation problem for $\Sigma_1$-Frege (i.e. resolution) is solvable in polynomial time, our result is close to optimal up to the computational complexity of solving mean-payoff games. The proof of the main result requires building low-depth formulas that compute the bits of the sum of a constant number of integers in binary notation, and low-complexity proofs of the required arithmetic properties.

## 1 Introduction

A mean-payoff game is played on a weighted directed graph $G = (V, E)$ with an integer weight $w(e)$ on every arc $e \in E$. Starting at an arbitrary vertex $u_0$, players 0 and 1 alternate in rounds, each extending the path $u_0, u_1, \ldots, u_n$ built up to that point, by adding one more arc $(u_n, u_{n+1}) \in E$ that leaves the current vertex $u_n$. The goal of player 0 is to maximize the long-run smallest average weight $\nu_0 = \liminf_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} w(u_{i-1}, u_i)$, while the goal of player 1 is to minimize the long-run largest average weight $\nu_1 = \limsup_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} w(u_{i-1}, u_i)$.

These games were studied by Ehrenfeucht and Mycielsky [13] who showed that every such game $\mathcal{G}$ has a value $\nu = \nu_{\mathcal{G}}$ such that player 0 has a *positional* strategy that secures $\nu_0 \geq \nu$, and player 1 has a *positional* strategy that secures $\nu_1 \leq \nu$. Here, a positional strategy is one whose moves depend only on the current vertex and not on the history of the play. We say that the game satisfies *positional determinacy.*

Positional determinacy is a property of interest in complexity theory. On one hand it implies that the problem of deciding if a given game has non-negative value (MPG) belongs to NP ∩ co-NP. This follows from the fact that every positional strategy has a short description, and that given a positional strategy for one player, it is possible to determine the best response strategy for the other in polynomial time. The latter was observed by Zwick and Paterson [29] as an application of

[*] A preliminary version of this paper appeared in the Proceedings of the 37th International Colloquium on Automata, Languages and Programming (ICALP 2010).

[†] Supported in part by CICYT TIN2007-68005-C04-03 (LOGFAC-2).

[‡] Supported in part by MICINN Ramon y Cajal and CICYT TIN2007-66523 (FORMALISM).

ISSN 1433-8092

Karp's algorithm for finding the minimum cycle mean in a digraph [17]. See [29] also for a direct link with Shapley's simple stochastic games. On the other hand, at the time of writing there is no known polynomial-time algorithm for solving mean-payoff games, not even for a special case called parity games that is of prime importance in applications of automata theory, and the body of literature on the topic keeps growing [16, 15, 9].

For a problem in NP $\cap$ co-NP for which a polynomial-time algorithm is not known or obvious, it is compulsory to ask for the nature of the certificates (short proofs of membership) and of the disqualifications (short proofs of non-membership). Celebrated examples where this was insightful are too many to be cited here (see [20, 22]). In the case that concerns us, that of mean-payoff games, a new and useful understanding of its membership in NP $\cap$ co-NP emerges from the combination of two recent results.

The starting point is the observation that the problem MPG reduces to the satisfiability problem for sets of *max-atoms*. A max-atom is an inequality of the form $x_0 \leq \max \{x_1 + a_1, \ldots, x_r + a_r\}$ where $x_0, \ldots, x_r$ are integer variables, and $a_1, \ldots, a_r$ are integer constants. This was first seen in [21] in the special context of scheduling and precedence constraints (with slightly different notation and definitions). The second result is from [8], where the satisfiability problem for max-atoms was re-discovered and given its name, and the problem was studied from the perspective of logic. The authors of [8] introduced an inference system, called *chaining*, that derives new max-atoms that follow from previous ones by simple rules. They showed that this system is both complete and, interestingly, polynomially bounded: if the collection of max-atom inequalities is unsatisfiable, then it has a refutation whose total size is polynomial in the size of the input.

Given these two results, the situation is that for a given mean-payoff game $\mathcal{G}$, a satisfying assignment to the corresponding instance of the max-atom problem is a certificate that $\nu_G \geq 0$, and a refutation of this instance in the chaining inference system is a certificate that $\nu_G < 0$. Therefore MPG reduces to the proof-search problem for this inference system. We address the question whether it also reduces to the proof-search problem for some standard proof-system for propositional logic. In brief, our main result is that a Boolean encoding of the instance expressing $\nu_{\mathcal{G}} \geq 0$ is either satisfiable, or has polynomial-size refutations in $\Sigma_2$-Frege, the standard inference system for propositional logic restricted to manipulating DNF-formulas. To be placed in context, in our terminology $\Sigma_1$-Frege manipulates clauses and is thus equivalent to propositional resolution.

**Related work and consequences.** The proof-search problem for a proof system $P$ asks, for a given unsatisfiable Boolean formula $A$, to find a $P$-refutation of $A$. We say that $P$ is automatizable if the proof-search problem for $P$ is solvable in time polynomial in the size of the smallest $P$-proof of $A$. The weak automatizability problem for $P$ asks, for a given formula $A$ and an integer $r$ given in unary, to distinguish the case when $A$ is satisfiable from the case when $A$ has a $P$-refutation of size at most $r$. It is known that this problem is solvable in polynomial time if and only if there is an automatizable proof system that simulates $P$.

The question whether some standard proof system is automatizable was introduced in [12], following the work in [19]. These works showed that extended-Frege and its weaker version TC$^0$-Frege are not automatizable unless there is a polynomial-time algorithm for factoring. Extended-Frege and TC$^0$-Frege are the standard inference systems for propositional logic restricted to manipulating Boolean circuits and threshold formulas of bounded depth, respectively. Indeed, their result is stronger since in both cases it shows that there is a reduction from factoring to the weak automatizability problem. To date, the weakest proof system that seems not weakly automatizable is

AC$^0$-Frege, the standard system restricted to Boolean formulas of bounded alternation-depth. But here the hardness result is much weaker since the reduction from factoring is only subexponential and degrades with the target depth of the AC$^0$-formulas [10].

All these hardness results proceed by exhibiting short refutations of an unsatisfiable Boolean formula that comes from a cryptography-inspired problem based on the hardness of factoring. Since the usual cryptographic primitives require either complex computations or complex proofs of correctness, going below polynomial-size TC$^0$-Frege or subexponential-size AC$^0$-Frege is difficult. In particular, there is no clear evidence in favour or against whether $\Sigma_d$-Frege, for fixed $d \geq 1$, is weakly automatizable, where $\Sigma_d$-formulas are AC$^0$-formulas of alternation-depth $d-1$ and a disjunction at the root. Not even for $\Sigma_1$-Frege (i.e. resolution) there is clear consensus in favour or against it, despite the partial positive results in [7, 5] and the partial negative results in [1].

The first consequence of our result is that the problem of solving mean-payoff games reduces to the weak-automatizability of $\Sigma_2$-Frege. Our initial goal was to reduce it to the weak-automatizability of resolution, or cutting planes, but these remain open. Note that cutting planes is a natural candidate in the context of max-atoms as it works with linear inequalities over the integers. The difficulty seems to be in simulating disjunctions of inequalities.

A second consequence of our result concerns the problem of interpolation for a proof system $P$. This is the problem that asks, for a given $P$-refutation of an unsatisfiable formula of the form $A_0(x, y_0) \wedge A_1(x, y_1)$ and a given truth assignment $a$ for $x$, to return an $i \in \{0, 1\}$ such that $A_i(a, y_i)$ is itself unsatisfiable. If the feasible interpolation problem for $P$ is solvable in polynomial time we say that $P$ enjoys feasible interpolation. It is known that feasible interpolation is closely related to weak automatizability in the sense that if a system is weakly automatizable, then it enjoys feasible interpolation [12, 27]. Proof systems enjoying feasible interpolation include resolution [18], cutting planes [24, 11], Lovász-Schrijver [26], and Hilbert's nullstellensatz [28]. On the negative side, it turns out that all known negative results for weak automatizability mentioned above were shown by reducing factoring to the interpolation problem. Thus, extended-Frege, TC$^0$-Frege and AC$^0$-Frege probably do not enjoy feasible interpolation. For $\Sigma_d$-Frege for fixed $d \geq 2$ there is no evidence in favour or against.

In this front our result implies that the problem of solving mean-payoff games reduces to the interpolation problem for $\Sigma_{2,2}$-Frege, where $\Sigma_{2,2}$-formulas are $\Sigma_3$-formulas of bottom fan-in two. Note that $\Sigma_1$-Frege does enjoy feasible interpolation since it is equivalent to resolution. Thus our result is close to optimal up to the computational complexity of solving mean-payoff games.

**Overview of the proof.** Given a mean-payoff game $\mathcal{G}$, we want to find an efficient translation of its associated instance of the max-atom problem into a collection of Boolean clauses. Once this is done, and assuming $\nu_G < 0$, we provide a polynomial-size $\Sigma_2$-Frege refutation that simulates the polynomial-size chaining-refutation guaranteed to exist by the results in [8].

Executing this plan requires technical work and is the main contribution of this paper. As part of its solution we need efficient depth-two formulas that compute the bits of the sum of a constant number of non-negative integers represented in binary. This was long known for two summands but the extension to more than two summands is not obvious and appears to be new. This turned out to be specially delicate because we need formulas explicit enough to allow polynomial-size depth-two Frege proofs of their basic properties. For example:

$$\frac{x \leq y + a \qquad y \leq z + b}{x \leq z + a + b}.$$

We hope these will be useful in independent contexts. One key fact in our argument is that we use the above with *constants* $a$ and $b$, which makes the bottom formula equivalent to $x \leq z + (a + b)$. The point is that if $a$ and $b$ were not constants, the number of summands would grow unbounded, and such sums are known to be not definable by polynomial-size formulas of constant depth [14].

**Structure of the paper.** In Section 2 we discuss the transformation from mean-payoff games to the max-atom problem, and the chaining inference system. In Section 3 we introduce the notation about Boolean formulas and the definition of $\Sigma_d$-Frege. In Section 4 we define the formula $\mathrm{CARRY}(x_1, \ldots, x_r)$ that computes the carry-bit of the sum of $r$ integers given in binary. In Section 6 we simulate the rules of chaining using formal proofs for the arithmetic properties of CARRY. In Section 7 we put everything together and get consequences for proof complexity.

## 2 Max-atom refutations

In this section we discuss the translation from mean-payoff games to the satisfiability problem for max-atom inequalities. We also define the chaining inference system and state its main property.

### 2.1 From mean-payoff games to max-atom inequalities

Let $\mathcal{G} = (V, E, V_0, V_1, w)$ be a mean-payoff game, which means that $(V, E)$ is a directed graph with out-degree at least one, $V = V_0 \cup V_1$ is a partition of the vertices into 0-vertices and 1-vertices, and $w : E \to \{-W, \ldots, 0, \ldots, W\}$ is an integer weight-assignment to the arcs of the graph. This specifies an instance of the mean-payoff game problem which asks whether $\nu \geq 0$. Here, $\nu = \min_{u \in V} \nu(u)$ and $\nu(u)$ is the value of the game started at $u$. This is defined as $\nu(u) = \sup_{s_0} \inf_{s_1} \nu(u, s_0, s_1)$, where $s_0$ and $s_1$ are strategies for player 0 and player 1, and

$$\nu(u, s_0, s_1) = \liminf_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} w(u_{i-1}, u_i)$$

where $u_0 = u$ and $u_{i+1} = s_j(u_0, \ldots, u_i)$ if $u_i \in V_j$ for $j \in \{0, 1\}$.

To every mean-payoff game $\mathcal{G}$ we associate a collection of max-atom inequalities $I(\mathcal{G})$ that is satisfiable if and only if $\nu \geq 0$. This was done for the first time in [21, Lemma 7.5]. Here we give a similar construction discussed in [4].

For every $u \in V$, we introduce one integer variable $x_u$. For every $u \in V_0$, we add

$$x_u \leq \max \{x_v + w(u, v) : v \in N(u)\},$$

where $N(u)$ is the set of out-neighbors of $u$ in $G$. For every $u \in V_1$, we want to impose the constraint

$$x_u \leq \min \{x_v + w(u, v) : v \in N(u)\}.$$

If $N(u) = \{v_1, \ldots, v_h\}$ this is simply

$$\begin{array}{rcl} x_u & \leq & \max \{x_{v_1} + w(u, v_1)\} \\ & \vdots & \\ x_u & \leq & \max \{x_{v_h} + w(u, v_h)\}. \end{array}$$

Note that $I(\mathcal{G})$ consists of at most $|E|$ max-atoms involving $|V|$ variables and integer constants in the range $[-W, W]$. Its size is thus polynomial in the size of $\mathcal{G}$. At this point we transformed the question whether $\nu \geq 0$ to the satisfiability of a system of max-atom inequalities. The correctness of the transformation is stated in Lemma 1 below.

## 2.2 Chaining refutations

An offset is a term of the form $x + c$, where $x$ is an integer variable and $c$ is an integer constant. In the following, the letters $R$ and $S$ refer to collections of offsets. Also, if $a$ is an integer constant, $S + a$ refers to the collection of offsets of the form $x + (c + a)$ as $x + c$ ranges over all offsets in $S$. The inference system introduced in [8] called *chaining* works with max-atom inequalities and has three rules. The first rule is called chaining:

$$\frac{x \leq \max(R, y + a) \qquad y \leq \max(S)}{x \leq \max(R, S + a)}.$$

The second rule is called simplification:

$$\frac{x \leq \max(R, x + a)}{x \leq \max(R)} \quad \text{if } a < 0.$$

The third rule is called contraction:

$$\frac{x \leq \max(R, y + a, y + b)}{x \leq \max(R, y + c)} \quad \text{if } a \leq c \text{ and } b \leq c.$$

A chaining refutation is a proof of $x \leq \max()$, which is clearly unsatisfiable.

This inference system is sound and complete for refuting unsatisfiable collections of max-atom inequalities [8, Theorem 2]. Even more, it is *polynomially bounded*, which means that if $\mathcal{I}$ is an unsatisfiable collection of max-atoms, then there is a chaining refutation of length polynomial in the size of $\mathcal{I}$, and with numbers of bit-length polynomial in the size of $\mathcal{I}$. This follows from two facts: that if $\mathcal{I}$ is unsatisfiable then it contains an unsatisfiable subcollection where every variable appears at most once on the left-hand side (Lemma 5 in [8]), and that for such subcollections the refutation produced by the completeness proof is polynomial (see the proof of Theorem 4 in [8]).

The following lemma states the correctness of the translation $I(\mathcal{G})$ and puts it together with what we need about max-atoms and chaining refutations:

**Lemma 1.** *Let $\mathcal{G} = (V, E, V_0, V_1, w)$ be a mean-payoff game and let $\mathcal{I} = I(\mathcal{G})$ be its transformation to a system of max-atom inequalities. Let $n = |V|$ and $m = |E|$, and $W = \max\{|w(e)| : e \in E\}$. The following are equivalent:*

1. *$\nu_{\mathcal{G}} < 0$,*
2. *$\mathcal{I}$ is unsatisfiable,*
3. *$\mathcal{I}$ is not satisfied by any assignment with values in the range $[0, mW]$,*
4. *$\mathcal{I}$ has a chaining refutation,*
5. *$\mathcal{I}$ has a chaining refutation of length at most $n^2$ with constants in the range $[-mW, mW]$.*

*Proof.* The equivalence between *1* and *2* follows (essentially) from Lemma 7.5 in [21] (see also [4] for a proof of the exact statement). The equivalence between *2* and *3* follows from Lemma 2 in [8]. The one between *3* and 4 follows from Theorem 2 in [8]. And the one between *4* and *5* follows from the remarks preceeding the statement of the lemma. □

## 3 Preliminaries in propositional logic

We introduce the notation and conventions related to Boolean formulas. We also define propositional proofs and discuss complexity measures. Besides these definitions, we also establish a few *schema* that will help us abbreviate the construction of proofs in later sections. Most of the concepts and notations in this section are standard in propositional proof complexity (see [25] or [6]).

### 3.1 Boolean formulas

Let $x_1, x_2, \ldots$ be a supply of Boolean variables. A literal is either a variable $x_i$, or the negation of a variable which we denote by $\overline{x_i}$, or the constant 1, or the negation of 1 which we denote by 0. We use literals to build Boolean formulas with the usual connectives: conjunctions $\wedge$ and disjunctions $\vee$. We think of conjunctions and disjunctions as associative, commutative and idempotent and therefore as symmetric connectives of unbounded arity.

If $A$ is a set of formulas, we write $\wedge A$ for the formula that joins all formulas in $A$ by a conjunction of arity $|A|$ at the root. Similarly, $\vee A$ denotes the formula that joins all formulas in $A$ by a disjunction of arity $|A|$ at the root. It will be convenient to allow negations on variables only. Thus, we think of $\neg \wedge A$ and $\neg \vee A$ as the same formulas as $\vee \neg A$ and $\wedge \neg A$, where $\neg A$ denotes the set of negations of formulas in $A$. When we reach the literals at the leaves, $\neg x_i$ denotes $\overline{x_i}$ and $\neg \overline{x_i}$ denotes $x_i$. If $F$ is a literal $x_i$ or $\overline{x_i}$, its size $s(F)$ is 1. If $F$ is a conjunction $\wedge A$ or a disjunction $\vee A$, its size $s(F)$ is $1 + \sum_{G \in A} s(G)$.

When writing formulas in text we use parenthesis to disambiguate different possible parse-trees. For example $F \vee G \wedge H$ has two possible parse-trees: $F \vee (G \wedge H)$ and $(F \vee G) \wedge H$. If $F(1), \ldots, F(r)$ denote formulas, when it is convenient we use the notation

$$\begin{aligned} (\forall i : 1 \leq i \leq r)(F(i)) &\equiv F(1) \wedge \cdots \wedge F(r), \\ (\exists i : 1 \leq i \leq r)(F(i)) &\equiv F(1) \vee \cdots \vee F(r). \end{aligned}$$

A clause is a disjunction of literals $\ell_1 \vee \cdots \vee \ell_r$. A term is a conjunction of literals $\ell_1 \wedge \cdots \wedge \ell_r$. A formula in CNF is a conjunction of clauses $C_1 \wedge \cdots \wedge C_m$. A formula in DNF is a disjunction of terms $T_1 \vee \cdots \vee T_m$. We define a hierarchy of formulas as follows: let $\Sigma_0 = \Pi_0$ be the set of all literals, and for $d \geq 1$, let $\Sigma_d$ be the collection of all formulas of the form $\vee A$, where $A$ is a set of $\Pi_{d-1}$-formulas, and let $\Pi_d$-formula be the collection of all formulas of the form $\wedge A$, where $A$ is a set of $\Sigma_{d-1}$-formulas. We write $\Sigma_{d,k}$ and $\Pi_{d,k}$ for the collection of all $\Sigma_{d+1}$- and $\Pi_{d+1}$-formulas with bottom fan-in at most $k$. For example, $\Sigma_{1,k}$ are $k$-DNF-formulas, that is, DNF-formulas composed of terms with at most $k$ literals. We use the notation $\Sigma_{d,\mathsf{c}}$ to denote $\Sigma_{d,k}$ for some unspecified constant $k \geq 1$.

## 3.2 Propositional proofs

We define four rules of inference. The four rules are axiom (AXM), weakening (WKG), introduction of conjunction (IOC), and cut (CUT):

$$\frac{}{F \vee \neg F} \qquad \frac{\Delta}{\Delta \vee G} \qquad \frac{\Delta \vee F \qquad \Delta' \vee G}{\Delta \vee \Delta' \vee (F \wedge G)} \qquad \frac{\Delta \vee F \qquad \Delta' \vee \neg F}{\Delta \vee \Delta'},$$

where $F$ and $G$ denote formulas, and $\Delta$ and $\Delta'$ denote either formulas or the special empty formula which we denote by $\square$. If $\Delta$ is the special empty formula, then $\Delta \vee \Delta'$ is simply $\Delta'$. Note that when $\Delta$ and $\Delta'$ are clauses and $F$ is a variable, the CUT-rule is also known as the resolution rule.

Let $F_1, \ldots, F_r$ and $G$ be formulas. The assertion that given $F_1, \ldots, F_r$ we can conclude $G$ is denoted by $F_1, \ldots, F_r \vdash G$. A proof of this assertion is a finite sequence of formulas $H_1, H_2, \ldots, H_m$ such that $H_m = G$ and for every $i \in [m]$, either $H_i = F_j$ for some $j \in [r]$, or $H_i$ is the conclusion of an inference rule with hypothesis $H_j$ and $H_k$ for some $j$ and $k$ such that $1 \leq j \leq k \leq i-1$. The length of the proof is $m$. The size of the proof is the sum of the sizes of all involved formulas. A refutation of $F_1, \ldots, F_r$ is a proof of the assertion $F_1, \ldots, F_r \vdash \square$. If $\mathcal{C}$ is a collection of formulas, a $\mathcal{C}$-Frege proof is one where all formulas belong to $\mathcal{C}$.

Whenever we use the expression "the assertion $F_1, \ldots, F_r \vdash G$ has a polynomial-size $\mathcal{C}$-Frege proof", what we mean is that there exists some universal but unspecified polynomial $p(n)$ such that $F_1, \ldots, F_r \vdash G$ has a $\mathcal{C}$-Frege proof of size at most $p(s(F_1) + \cdots + s(F_r) + s(G))$. Similarly, we use $\texttt{poly}(n)$ to denote some universal but unspecified polynomial function of $n$, and $\texttt{c}$ to denote some universal but unspecified constant.

A resolution proof is one where all formulas are clauses and the only allowed rule is CUT. Note that if the only allowed formulas are clauses then IOC is automatically forbidden. Also it is not hard to see that using the rules AXM and WKG makes no difference when only clauses are used: if there is a $\Sigma_1$-Frege refutation of $F_1, \ldots, F_r$ of length $m$, then there is a resolution refutation of $F_1, \ldots, F_r$ of length at most $m$ as well. Therefore resolution and $\Sigma_1$-Frege are essentially the same thing. Let us mention that $\Sigma_{1,k}$-Frege is also known as Res($k$), or as $k$-DNF-resolution. Along these lines, $\Sigma_2$-Frege could be called DNF-resolution.

## 3.3 Proof schema

A *proof scheme* is a statement saying that a formal proof of a certain assertion $\alpha_1, \ldots, \alpha_r \vdash \beta$ can be converted to a proof of a related assertion $\alpha'_1, \ldots, \alpha'_s \vdash \beta'$. In this section we provide three proof schema for later use.

**Proof-scheme of weakening** The first proof-scheme states that if there exists a small proof of an assertion with two hypothesis, then there exists a small proof of the same assertion where one of its hypothesis and the conclusion have been weakened by the addition of a disjunct. For later applications, we need to be particularly careful with the size and the length of the resulting proof.

**Lemma 2.** *Let $W$ be a $\Sigma_{d,k}$-formula of size at most $t$. If*

$$\frac{F \qquad G}{H}$$

*has a proof of length $\ell$ with $\Sigma_{d,k}$-formulas of size at most $s$, then*

$$\frac{F \qquad G \vee W}{H \vee W}$$

*has a proof of length at most $\ell + 1$ with $\Sigma_{d,k}$-formulas of size at most $s + t + 1$.*

*Proof.* Replace the hypothesis $G$ by $G \vee W$ and apply the same rules as in the given proof. The side formula $W$ accumulates along the proof to produce the conclusion $H \vee W$, or $H$ if the right hypothesis is really not used. In the second case just add $W$ by weakening. The length of the new proof is at most $\ell + 1$. For the size, in the worst case $W$ appears as a side formula of each line of the new proof. Therefore each line increases its size by at most $t + 1$ (the $+1$ takes care of the potentially new disjunction-node at the root). □

**Proof-scheme of pairwise case-analysis** We continue with a proof-scheme showing that in order to have a small proof of $H$ from the two hypothesis $(\exists i)(F(i))$ and $(\exists j)(G(j))$, it is enough to have small proofs of $H$ from each particular pair of hypothesis $F(a)$ and $G(b)$, for all possible values of $a$ and $b$.

**Lemma 3.** *If for every $a \in [r]$ and every $b \in [s]$ the assertion*

$$\frac{F(a) \qquad G(b)}{H}$$

*has a proof of length at most $\ell$ with $\Sigma_{d,k}$-formulas of size at most $t$, then*

$$\frac{(\exists i : 1 \leq i \leq r)(F(i)) \qquad (\exists j : 1 \leq j \leq s)(G(j))}{H}$$

*has a proof of length at most $\texttt{poly}(r, s, \ell)$ with $\Sigma_{d,k}$-formulas of size at most $\texttt{poly}(r, s, t)$.*

*Proof.* For $a \in [r]$ and $b \in [s]$, let (H.$a$.$b$) denote the assertion in the hypothesis. We start giving a proof of the assertion

$$\frac{(\exists i : 1 \leq i \leq a)(F(i)) \qquad G(b)}{H} \tag{1}$$

for every fixed $a \in [r]$ and $b \in [s]$. To achieve this we fix $b \in [s]$ and proceed inductively on $a \in [r]$. The base case is $a = 1$ in which case (1.$a$.$b$) is given by hypothesis since the formula $(\exists i : 1 \leq i \leq 1)(F(i))$ is a different way of writing $F(1)$. Assuming $a \in \{2, \ldots, r\}$ and that we have a proof of (1.$a-1$.$b$), we give a proof of (1.$a$.$b$). First apply the proof-scheme of weakening on the proof of (H.$a$.$b$) by adding $(\exists i : 1 \leq i \leq a-1)(F(i))$ to its left hypothesis and to the conclusion. This gives a proof of

$$\frac{(\exists i : 1 \leq i \leq a)(F(i)) \qquad G(b)}{(\exists i : 1 \leq i \leq a-1)(F(i)) \vee H}. \tag{2}$$

Then apply the proof-scheme of weakening on the proof of (1.$a-1$.$b$) by adding $H$ to its left hypothesis and to the conclusion. This gives a proof of

$$\frac{(\exists i : 1 \leq i \leq a-1)(F(i)) \vee H \qquad G(b)}{H}. \tag{3}$$

Concatenating the proof of (2.$a$.$b$) with that of (3.$a$.$b$) we get a proof of (1.$a$.$b$).

Before we continue, let us analyze the length and size of this proof. Let $L(a, b)$ be the length of the proof, and let $S(a, b)$ be the maximum size of the formulas in the proof. For $a = 1$, we have

$L(1,b) \leq \ell$ and $S(1,b) \leq t$ by hypothesis. For $a > 1$, from the estimates in the proof-scheme of weakening we get the following recurrences:

$$\begin{aligned} &L(a,b) \leq \ell + 1 + L(a-1,b) + 1 \\ &S(a,b) \leq t + (a-1)t + 1 + 1 + S(a-1,b) + t + 1. \end{aligned}$$

Expanding we get $L(r,b) \leq p(r,\ell)$ and $S(r,b) \leq q(r,t)$ for certain polynomials $p$ and $q$.

We continue giving a proof of

$$\frac{(\exists i : 1 \leq i \leq r)(F(i)) \qquad (\exists j : 1 \leq j \leq b)(G(j))}{H} \tag{4}$$

for every $b \in [s]$. To achieve this we proceed inductively on $b \in [s]$. The base case is $b = 1$ in which case $(4.b)$ is precisely $(1.r.1)$ because the formula $(\exists j : 1 \leq j \leq 1)(G(j))$ is a different way of writing $G(1)$. Assuming $b \in \{2, \ldots, s\}$ and that we have a proof of $(4.b-1)$, we give a proof of $(4.b)$. First apply the proof-scheme of weakening on the proof of $(1.r.b)$ by adding $(\exists j : 1 \leq j \leq b-1)(G(j))$ to its right hypothesis and to the conclusion. This gives a proof of

$$\frac{(\exists i : 1 \leq i \leq r)(F(i)) \qquad (\exists j : 1 \leq j \leq b)(G(j))}{(\exists j : 1 \leq j \leq b-1)(G(j)) \vee H}. \tag{5}$$

Then apply the proof-scheme of weakening on the proof of $(4.b-1)$ by adding $H$ to its right hypothesis and to the conclusion. This gives a proof of

$$\frac{(\exists i : 1 \leq i \leq r)(F(i)) \qquad (\exists j : 1 \leq j \leq b-1)(G(j)) \vee H}{H}. \tag{6}$$

Concatenating the proof of $(5.b)$ with that of $(6.b)$ we get a proof of $(4.b)$.

We conclude with the analysis of the length and the size of this proof. Let $L(b)$ be the length of the proof and let $S(b)$ be the maximum size of the formulas in the proof. For $b = 1$, we have $L(1) \leq p(r,\ell)$ and $S(1) \leq q(r,t)$ from the analysis of the proof of $(1.a.b)$. For $b > 1$, from the estimates in the proof-scheme of weakening we get the following recurrences:

$$\begin{aligned} &L(b) \leq p(r,\ell) + 1 + L(b-1) + 1 \\ &S(b) \leq q(r,t) + (b-1)q(r,t) + 1 + 1 + S(b-1) + q(r,t) + 1. \end{aligned}$$

Expanding we get $L(r,s) \leq p'(r,s,\ell)$ and $S(r,s) \leq q'(r,s,t)$ for certain polynomials $p'$ and $q'$. □

**Proof-scheme of case-analysis** The next proof-scheme is a particular case of the previous one. For later reference we state it as a lemma.

**Lemma 4.** *If for every $a \in [r]$ the assertion*

$$\frac{F(a)}{H}$$

*has a proof of length at most $\ell$ with $\Sigma_{d,k}$-formulas of size at most $t$, then*

$$\frac{(\exists i : 1 \leq i \leq r)(F(i))}{H}$$

*has a proof of length* $\mathtt{poly}(r,\ell)$ *with $\Sigma_{d,k}$-formulas of size* $\mathtt{poly}(r,t)$.

*Proof.* This is a very special case of the proof-scheme of pairwise case-analysis where $r = s$ and $F(i) = G(i)$ for every $i \in \{1, \ldots, r\}$. □

**Scheme of implication** The following lemma says that there is a small proof of an existential-universal formula of the form $(\exists i)(\forall j)(G(i,j))$ starting from the hypothesis $(\exists i)(\forall j)(F(i,j))$ and all the implications $F(a,b) \to G(a,b)$.

**Lemma 5.** *For every $i \in [r]$ and $j \in [s]$, let $F(i,j)$ be a $\Pi_{1,k}$-formula and let $G(i,j)$ be a $\Sigma_{1,k}$-formula. The following assertion has a polynomial-size $\Sigma_{2,k}$-Frege proof:*
*Given*

1. $\neg F(a,b) \vee G(a,b)$ *for every* $a \in [r]$ *and* $b \in [s]$,
2. $(\exists i : 1 \leq i \leq r)(\forall j : 1 \leq j \leq s)(F(i,j))$,

*conclude*

$$(\exists i : 1 \leq i \leq r)(\forall j : 1 \leq j \leq s)(G(i,j)).$$

*Proof.* For every $a \in [r]$ and $b \in [s]$, let (H.$a,b$) denote the hypothesis numbered *1.* for the indicated values of $a$ and $b$. Let (H) denote the hypothesis numbered *2.* For every fixed $a \in [r]$, apply IOC on (H.$a$.1), ..., (H.$a$.$s$) to get

$$\neg F(a,1) \vee \cdots \vee \neg F(a,s) \vee (\forall j : 1 \leq j \leq s)(G(a,j)). \tag{7}$$

Note that $\neg F(a,1) \vee \ldots \vee \neg F(a,s)$ is the negation of $(\forall j : 1 \leq j \leq s)(F(a,j))$. Apply CUT between $(H)$ and (7.1) on this formula for $a = 1$, followed by CUT between the result and (7.2) on the same formula for $a = 2$, and so on until $a = r$. This gives

$$(\forall j : 1 \leq j \leq s)(G(1,j)) \vee \cdots \vee (\forall j : 1 \leq j \leq s)(G(r,j)) \tag{8}$$

which is exactly the goal. □

# 4 Bitwise linear arithmetic

The basic $\Sigma_{2,\mathsf{c}}$-formula with which we work expresses an inequality. More specifically, it asserts that an addition results in "overflow", or equivalently that there is a carry-bit generated at the left most position. As a simple example, suppose we want to express that the sum of two $B$-bit numbers $x = x_1 \ldots x_B$ and $y = y_1 \ldots y_B$ is at least $2^B$. It is not hard to see that the following formula is equivalent to the desired inequality:

$$(\exists p : 1 \leq p \leq B)(x_p = 1 \wedge y_p = 1 \wedge (\forall q : 1 \leq q \leq p-1)(x_q + y_q = 1)).$$

By writing $x_q + y_q = 1$ in conjunctive normal form, note that this is a $\Sigma_{2,2}$-formula. In this section we generalize this formula to an arbitrary number of $B$-bit numbers.

## 4.1 Automaton and formula

Let $r$, $k$, $\ell$ and $B$ be positive integers such that $r \leq k \leq 2^\ell - 1 < 2^B$. Let $\mathbf{x} = (x_1, \ldots, x_r)$, where each $x_i$ is a string $x_{i,1} \ldots x_{i,B}$ of $B$ Boolean variables. We think of $\mathbf{x}$ as a matrix with $r$ rows and $B$ columns arranged as follows:

$$\mathbf{x} = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,B} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,B} \\ \vdots & \vdots & \ddots & \vdots \\ x_{r,1} & x_{r,2} & \cdots & x_{r,B} \end{pmatrix}$$

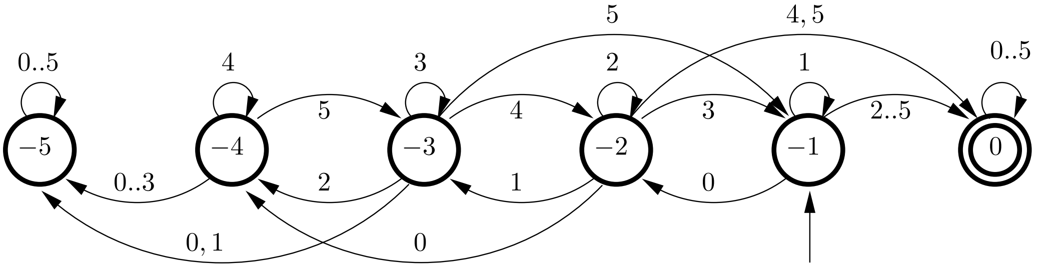

Figure 1: A state machine that decides if there is overflow in the addition of 5 Boolean strings.

For each column $p \in \{1, \ldots, B\}$, let $\mathbf{x}_p = x_{1,p} + \cdots + x_{r,p}$. We interpret $\mathbf{x}_p$ as a symbol in an alphabet of $r+1$ symbols $\{0, \ldots, r\} \subseteq \{0, \ldots, k\}$, and thus $\mathbf{x}$ as a word in $\{0, \ldots, k\}^B$.

We describe an automaton $M$ that decides whether there is overflow in the addition of $r$ $B$-bit numbers. It is defined to work on the alphabet $\{0, 1, \ldots, k\}$, i.e. its input is $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_B$. In general, $M$ has $k+1$ states each indicating a range for the value of the number read so far, which at step $p$ we will denote by $\mathbf{x}_{[p]} = \mathbf{x}_1 2^{p-1} + \mathbf{x}_2 2^{p-2} + \cdots + \mathbf{x}_{p-1} 2^1 + \mathbf{x}_p 2^0$. The $k+1$ states correspond to the ranges $[0, 2^p - k]$, the following $k-1$ single integer intervals $2^p - (k-1), \ldots, 2^p - 1$, and $[2^p, k(2^p - 1)]$. We denote these states by $-k$, $-(k-1), \ldots, -1$, and $0$, respectively. The two extreme states are absorbing, and correspond respectively to the absence and presence of overflow: if $\mathbf{x}_{[p]} \geq 2^p$ then $\mathbf{x} \geq 2^p 2^{B-p} = 2^B$, hence there is overflow; on the other hand, if $\mathbf{x}_{[p]} \leq 2^p - k$ then $\mathbf{x} \leq (2^p - k)2^{B-p} + k(2^{B-p} - 1) = 2^B - 1$, and there is no overflow. The starting state is $-1$, because $\mathbf{x}_{[0]} = 0 = 2^0 - 1$. The state machine for $k = 5$ is given in Figure 1.

The key fact that allows us to design the propositional formula is that if at some stage the machine has not yet reached one of the absorbing states, then we can identify in which intermediate state it is only based on the last $\ell$ values read, because it suffices to know $\mathbf{x}_{[p]}$ modulo $2^\ell > k - 1$.

We define notation for the number in the last $\ell$ positions, and the state it corresponds to:

- $A_\ell(\mathbf{x}; p) = \mathbf{x}_1 2^{p-1} + \mathbf{x}_2 2^{p-2} + \cdots + \mathbf{x}_{p-1} 2^1 + \mathbf{x}_p 2^0$ if $0 \leq p \leq \ell - 1$,
- $A_\ell(\mathbf{x}; p) = \mathbf{x}_{p-\ell+1} 2^{\ell-1} + \mathbf{x}_{p-\ell+2} 2^{\ell-2} + \cdots + \mathbf{x}_{p-1} 2^1 + \mathbf{x}_p 2^0$ if $\ell \leq p \leq B$,
- $S_\ell(\mathbf{x}; p) = (A_\ell(\mathbf{x}; p) \bmod 2^p) - 2^p$ if $0 \leq p \leq \ell - 1$,
- $S_\ell(\mathbf{x}; p) = (A_\ell(\mathbf{x}; p) \bmod 2^\ell) - 2^\ell$ if $\ell \leq p \leq B$,
- $N_\ell(\mathbf{x}; p) = 2S_\ell(\mathbf{x}; p-1) + \mathbf{x}_p$ if $1 \leq p \leq B$.

Intuitively, $S_\ell(\mathbf{x}; p)$ denotes the state of the computation of $M$ at time $p$ as long as it did not reach an absorbing state before, and $N_\ell(\mathbf{x}; p)$ stands for "next state" when position $p$ is read even though it is not always in the range $\{-k, \ldots, 0\}$.

For every $p \in \{1, \ldots, B\}$, we define the predicates

$$\begin{aligned} F^+(\mathbf{x}; p) &\equiv F^+_{k,\ell}(\mathbf{x}; p) \equiv N_\ell(\mathbf{x}; p) \geq 0, \\ F^-(\mathbf{x}; p) &\equiv F^-_{k,\ell}(\mathbf{x}; p) \equiv -k < N_\ell(\mathbf{x}; p) < 0. \end{aligned}$$

When the parameters $k$ and $\ell$ are clear from the context we use the lighter notation on the left. Assuming that $S_\ell(\mathbf{x}; p-1)$ is the correct state of $M$ at time $p-1$, the predicate $F^+(\mathbf{x}; p)$ asserts that at time $p$ the automaton accepts, and $F^-(\mathbf{x};p)$ asserts that at time $p$ the automaton is not at an absorbing state.

Since $F^+(\mathbf{x};p)$ and $F^-(\mathbf{x};p)$ depend on no more than $k\ell$ variables of $\mathbf{x}$, those appearing in the definitions of $\mathbf{x}_{p-\ell+1},\dots,\mathbf{x}_p$, both $F^+(\mathbf{x};p)$ and $F^-(\mathbf{x};p)$ are expressible as $\Sigma_{1,k\ell}$-formulas and as $\Pi_{1,k\ell}$-formulas of size at most $k\ell \cdot 2^{k\ell}$. Using these we define the following formula:

$$\mathrm{CARRY}_{k,\ell}(\mathbf{x}) \quad\equiv\quad (\exists p : 1 \leq p \leq B)(F^+(\mathbf{x};p) \wedge (\forall q : 1 \leq q \leq p-1)(F^-(\mathbf{x};q)))$$

Intuitively, this formula reads "$M$ eventually accepts". Note that this is a $\Sigma_{2,k\ell}$-formula of size proportional to $B^2 \cdot k\ell \cdot 2^{k\ell}$.

## 4.2 A technical lemma

The following key lemma states that if the predicted next state $N_\ell$ is not absorbing, then it is correct. This will be used intensively in the next section.

**Lemma 6.** *Let $1 \leq p \leq B$. If $-k < N_\ell(\mathbf{z};p) < 0$, then $S_\ell(\mathbf{z};p) = N_\ell(\mathbf{z};p)$.*

*Proof.* Let $\ell' = \min\{p,\ell\}$ and $\ell'' = \min\{p-1,\ell\}$.

$$\begin{aligned} N_\ell(\mathbf{z};p) &\equiv 2S_\ell(\mathbf{z};p-1) + \mathbf{z}_p \mod 2^{\ell'} \\ &\equiv 2((A_\ell(\mathbf{z};p-1) \bmod 2^{\ell''}) - 2^{\ell''}) + \mathbf{z}_p \mod 2^{\ell'} \\ &\equiv 2(A_\ell(\mathbf{z};p-1) + m2^{\ell''} - 2^{\ell''}) + \mathbf{z}_p \mod 2^{\ell'} \end{aligned}$$

for some integer $m$. Since $2^{\ell''+1}$ is a multiple of $2^{\ell'}$, we infer

$$N_\ell(\mathbf{z};p) \equiv 2A_\ell(\mathbf{z};p-1) + \mathbf{z}_p \mod 2^{\ell'}.$$

Now note that

$$\begin{aligned} 2A_\ell(\mathbf{z};p-1) + \mathbf{z}_p &= A_\ell(\mathbf{z};p) && \text{if } p \leq \ell, \\ 2A_\ell(\mathbf{z};p-1) + \mathbf{z}_p &= A_\ell(\mathbf{z};p) + 2^\ell \mathbf{z}_{p-\ell} && \text{if } p > \ell. \end{aligned}$$

Since $2^\ell$ is a multiple of $2^{\ell'}$, in both cases we get

$$\begin{aligned} N_\ell(\mathbf{z};p) &\equiv A_\ell(\mathbf{z};p) \mod 2^{\ell'} \\ &\equiv (A_\ell(\mathbf{z};p) \bmod 2^{\ell'}) - 2^{\ell'} \mod 2^{\ell'} \\ &\equiv S_\ell(\mathbf{z};p) \mod 2^{\ell'}. \end{aligned}$$

This means that the residue classes of $S_\ell(\mathbf{z};p)$ and $N_\ell(\mathbf{z};p)$ are the same. At this point we need to distinguish the cases $p > \ell$ and $p \leq \ell$.

In case $p > \ell$ we have $\ell' = \ell$. Notice that $-2^{\ell'} \leq S_\ell(\mathbf{z};p) \leq -1$ and

$$-2^{\ell'} \leq -k+1 \leq N_\ell(\mathbf{z};p) \leq -1$$

by the assumption. Therefore, the congruence $S_\ell(\mathbf{z};p) \equiv N_\ell(\mathbf{z};p)$ is actually an equality $S_\ell(\mathbf{z};p) = N_\ell(\mathbf{z};p)$.

In case $p \leq \ell$ we have $\ell' = p$. Here we have $-2^{p-1} \leq S_\ell(\mathbf{z};p-1) \leq -1$. Therefore

$$-2^{\ell'} \leq 2S_\ell(\mathbf{z};p-1) \leq 2S_\ell(\mathbf{z};p-1) + \mathbf{z}_p = N_\ell(\mathbf{z};p) \leq -1$$

where the first inequality follows from the above, the second inequality follows from $\mathbf{z}_p \geq 0$, and the third inequality follows from the assumption. As in the previous case also $-2^{\ell'} \leq S_\ell(\mathbf{z};p) \leq -1$ and therefore the congruence $S_\ell(\mathbf{z};p) \equiv N_\ell(\mathbf{z};p)$ is actually an equality $S_\ell(\mathbf{z};p) = N_\ell(\mathbf{z};p)$. □

# 5 Proofs of arithmetic facts

In this section $k$, $\ell$ and $B$ are integers such that $k \leq 2^\ell - 1 < 2^B$. We think of $k$ and $\ell$ as small and bounded by some universal constant, and of $B$ as unbounded. For concreteness, the uncomfortable reader should fix $k = 11$ and $\ell = 4$ as we will do in later applications. In particular $\mathrm{CARRY}_{k,\ell}$ is a $\Sigma_{2,\mathsf{c}}$-formula, for some unspecified universal bottom fan-in, and the expression "polynomial-size $\Sigma_{2,\mathsf{c}}$-Frege proof" refers to a proof of size $\mathtt{poly}(B)$, for some unspecified universal polynomial.

The letters $a$, $b$ and $c$ denote $B$-bit strings $a_1 \dots a_B$, $b_1 \dots b_B$ and $c_1 \dots c_B$, respectively. Abusing a bit the notation, sometimes we identify the string $a$ with the number in $[0, 2^B)$ that it represents in binary. Similarly, we identify 0 and 1 with the strings $0^B$ and $0^{B-1}1$, respectively.

We distinguish two types of elementary facts: bookkeeping facts, where not much arithmetic is happening, and arithmetic facts, where the meat is.

## 5.1 Bookkeeping facts

In this subsection the letter $\mathbf{x}$ denotes a non-empty sequence $(x_1, \dots, x_r)$, where each $x_i$ is a string $x_{i,1} \dots x_{i,B}$ of $B$ Boolean variables. The letters $u$, $v$, $w$ and $y$ denote sequences of Boolean variables such that $|u| = |w|$, $|v| = |y|$, and $|u| + |v| = B - 1$.

**Lemma 7.** *The following assertions have polynomial-size $\Sigma_{2,\mathsf{c}}$-Frege proofs:*

1. *given* $\mathrm{CARRY}_{k,\ell}(\mathbf{x}, 0)$ *conclude* $\mathrm{CARRY}_{k,\ell}(\mathbf{x})$*, if* $r + 1 \leq k$*,*
2. *given* $\mathrm{CARRY}_{k,\ell}(\mathbf{x})$ *conclude* $\mathrm{CARRY}_{k,\ell}(\mathbf{x}, 0)$*, if* $r + 1 \leq k$*,*
3. *given* $\mathrm{CARRY}_{r,\ell}(\mathbf{x})$ *conclude* $\mathrm{CARRY}_{k,\ell}(\mathbf{x})$*, if* $r \leq k$*,*
4. *given* $\mathrm{CARRY}_{k,\ell}(\mathbf{x})$ *conclude* $\mathrm{CARRY}_{r,\ell}(\mathbf{x})$*, if* $r \leq k$*,*
5. *given* $\mathrm{CARRY}_{k,\ell}(\mathbf{x}, u0v, w1y)$ *conclude* $\mathrm{CARRY}_{k,\ell}(\mathbf{x}, u1v, w0y)$*, if* $r + 2 \leq k$*.*

**Proof of Lemma 7.1 and 7.2** For every $p \in \{1, \dots, B\}$, define formulas

$$\begin{array}{rcl} F^+(p) & \equiv & N_\ell(\mathbf{x}, 0; p) \geq 0, \\ G^+(p) & \equiv & N_\ell(\mathbf{x}; p) \geq 0, \\ F^-(p) & \equiv & -k < N_\ell(\mathbf{x}, 0; p) < 0, \\ G^-(p) & \equiv & -k < N_\ell(\mathbf{x}; p) < 0. \end{array}$$

Clearly $F^+(p)$ and $G^+(p)$ are equivalent. Similarly $F^-(p)$ and $G^-(p)$ are equivalent. This means that the following formulas are tautologies:

$$\neg F^+(p) \vee G^+(p)$$
$$\neg F^-(p) \vee G^-(p)$$
$$\neg G^+(p) \vee F^+(p)$$
$$\neg G^-(p) \vee G^-(p).$$

Since these are constant-size $\Sigma_{1,\mathsf{c}}$-formulas, by completeness they have $\Sigma_{1,\mathsf{c}}$-Frege proofs of constant size. The proof now follows from two applications of the scheme of implication Lemma 5.

**Proof of Lemma 7.3 and 7.4** For every $s \in \{1, \ldots, B\}$ define formulas

$$F^+(s) \equiv H^+(s) \equiv N_\ell(\mathbf{x}; s) \geq 0,$$

and

$$\begin{aligned} F^-(s) &\equiv -k < N_\ell(\mathbf{x}; s) < 0, \\ H^-(s) &\equiv -r < N_\ell(\mathbf{x}; s) < 0, \\ R(s) &\equiv N_\ell(\mathbf{x}; s) > -r. \end{aligned}$$

For every $s \in \{1, \ldots, B\}$ define formulas

$$\begin{aligned} F^*(s) &\equiv (\forall q : 1 \leq q \leq s-1)(F^-(q)), \\ H^*(s) &\equiv (\forall q : 1 \leq q \leq s-1)(H^-(q)). \end{aligned}$$

We start with the proof of

$$\frac{\mathrm{CARRY}_{k,\ell}(\mathbf{x})}{\mathrm{CARRY}_{r,\ell}(\mathbf{x})}. \tag{9}$$

From the definitions of the formulas and the fact that $r \leq k$, the following are tautologies for every fixed $p \in \{1, \ldots, B\}$:

$$\begin{gathered} \neg H^-(p) \vee F^-(p), \\ \neg H^+(p) \vee F^+(p). \end{gathered}$$

These are constant-size $\Sigma_{1,\mathsf{c}}$-formulas and therefore, by completeness, they have $\Sigma_{1,\mathsf{c}}$-Frege proofs of constant size. The scheme of implication Lemma 5 gives then (9). Next we give the proof of

$$\frac{\mathrm{CARRY}_{r,\ell}(\mathbf{x})}{\mathrm{CARRY}_{k,\ell}(\mathbf{x})}. \tag{10}$$

From the definitions of the formulas, the following are tautologies for every fixed $p \in \{1, \ldots, B\}$:

$$\neg F^-(p) \vee \neg R(p) \vee H^-(p), \tag{11}$$

$$\neg F^+(p) \vee H^+(p). \tag{12}$$

Additionally we argue the validity of the following for every fixed $p \in \{2, \ldots, B\}$:

$$\neg F^-(p-1) \vee \neg F^+(p) \vee R(p), \tag{13}$$

$$\neg F^-(p-1) \vee \neg R(p) \vee R(p-1). \tag{14}$$

The validity of (13.$p$) follows again directly from the definitions of the formulas. The validity of (14.$p$) follows from the next Claim.

**Claim 1.** *Let $2 \leq p \leq B$.*

*either $N_\ell(\mathbf{x}; p-1) \leq -k$, or $N_\ell(\mathbf{x}; p) \leq -r$, or $N_\ell(\mathbf{x}; p-1) > -r$.*

*Proof.* Assume $N_\ell(\mathbf{x}; p-1) \leq -r$ and $N_\ell(\mathbf{x}; p-1) > -k$. In particular $-k < N_\ell(\mathbf{x}; p-1) < 0$ and by Lemma 6 we have

$$S_\ell(\mathbf{x}; p-1) = N_\ell(\mathbf{x}; p-1). \tag{15}$$

Therefore

$$N_\ell(\mathbf{x}; p) = 2S_\ell(\mathbf{x}; p-1) + \mathbf{x}_p \leq -2r + \mathbf{x}_p \leq -r,$$

where the first inequality follows from (15) and the assumption that $N_\ell(\mathbf{x}; p-1) \leq -r$, and the second inequality follows from the fact that $\mathbf{x}_p \leq r$. □

We continue with the proof of (10). All of (11.$p$), (12.$p$), (13.$p$), and (14.$p$) are constant-size $\Sigma_{1,\mathsf{c}}$-formulas and therefore, by completeness, they have $\Sigma_{1,\mathsf{c}}$-Frege proofs of constant size. With these in hand we will derive a proof of the following assertion, for every fixed $p \in \{1, \ldots, B\}$:

$$\frac{F^*(p) \wedge F^+(p)}{\mathrm{CARRY}_{k,\ell}(\mathbf{x})}. \tag{16}$$

The proof-scheme of case-analysis Lemma 4 will do the rest to complete the proof.

The case $p = 1$ is obtained directly by a CUT between (12.$p$) and the hypothesis followed by weakening. For $p \in \{2, \ldots, B\}$ we start applying CUT between (13.$p$) and (14.$p$) to get $\neg F^-(p-1) \vee \neg F^+(p) \vee R(p-1)$. Then apply CUT between this and (14.$p-1$) to get $\neg F^-(p-2) \vee \neg F^-(p-1) \vee \neg F^+(p) \vee R(p-2)$. Continuing like this until we use (14.1) we get

$$\neg F^-(q) \vee \neg F^-(q+1) \vee \ldots \vee \neg F^-(p-1) \vee \neg F^+(p) \vee R(q) \tag{17}$$

for every $q \in \{1, \ldots, p-1\}$. Then CUT between (17.$q$) and (11.$q$) gives

$$\neg F^-(q) \vee \neg F^-(q+1) \vee \ldots \vee \neg F^-(p-1) \vee \neg F^+(p) \vee H^-(q). \tag{18}$$

At this point, IOC on (18.1), $\ldots$, (18.$p-1$) and (12.$p$), followed by CUT with the hypothesis and weakening, gives the goal in (16.$p$).

**Proof of Lemma 7.5** For every $p \in \{1, \ldots, B\}$, define formulas

$$\begin{array}{rcl} F^+(p) & \equiv & N_\ell(\mathbf{x}, u0v, w1y; p) \geq 0, \\ G^+(p) & \equiv & N_\ell(\mathbf{x}; u1v, w0y; p) \geq 0, \\ F^-(p) & \equiv & -k < N_\ell(\mathbf{x}, u0v, w1y; p) < 0, \\ G^-(p) & \equiv & -k < N_\ell(\mathbf{x}, u1v, w0y; p) < 0. \end{array}$$

The formulas $F^+(p)$ and $G^+(p)$ are equivalent. Similarly $F^-(p)$ and $G^-(p)$ are equivalent. This means that the following formulas are tautologies:

$$\begin{array}{c} \neg F^+(p) \vee G^+(p) \\ \neg F^-(p) \vee G^-(p). \end{array}$$

Since these are constant-size $\Sigma_{1,\mathsf{c}}$-formulas, by completeness they have $\Sigma_{1,\mathsf{c}}$-Frege proofs of constant size. The proof now follows from an application of the scheme of implication Lemma 5.

## 5.2 Arithmetic facts

In this subsection the letter $z$ denotes a string of $B$ Boolean variables $z_1 \ldots z_B$. We write $\overline{z}$ for the string of complementary literals: $\overline{z_1} \ldots \overline{z_B}$. The letters $\mathbf{x}$ and $\mathbf{y}$ denote non-empty sequences $(x_1, \ldots, x_{r_x})$ and $(y_1, \ldots, y_{r_y})$, where each $x_i$ is a string of $B$ Boolean variables $x_{i,1} \ldots x_{i,B}$ and each $y_i$ is a string of $B$ Boolean variables $y_{i,1} \ldots y_{i,B}$.

**Lemma 8.** *The following assertions have polynomial-size $\Sigma_{2,\mathsf{c}}$-Frege proofs:*

1. *given* $\mathrm{CARRY}_{k,\ell}(\mathbf{x}, z, 1)$ *and* $\mathrm{CARRY}_{k,\ell}(\mathbf{y}, \overline{z}, 1)$ *conclude* $\mathrm{CARRY}_{k,\ell}(\mathbf{x}, \mathbf{y}, 1)$*, if* $r_x + r_y + 1 \leq k$,
2. *given* $\mathrm{CARRY}_{k,\ell}(\mathbf{x}, a, b)$ *conclude* $\mathrm{CARRY}_{k,\ell}(\mathbf{x}, c)$*, if* $c = a + b$ *and* $r_x + 3 \leq k$,
3. *given* $\mathrm{CARRY}_{k,\ell}(\mathbf{x}, a)$ *conclude* $\mathrm{CARRY}_{k,\ell}(\mathbf{x}, b)$*, if* $a \leq b$ *and* $r_x + 2 \leq k$,
4. *given* $\mathrm{CARRY}_{k,\ell}(z, \overline{z})$ *conclude* $\square$.

**Proof of Lemma 8.1** Let $r = r_x + 2$ and $s = r_y + 2$. Applying Lemma 7.4 on the two hypothesis $\mathrm{CARRY}_{k,\ell}(\mathbf{x}, z, 1)$ and $\mathrm{CARRY}_{k,\ell}(\mathbf{x}, \overline{z}, 1)$ we obtain

$$\mathrm{CARRY}_{r,\ell}(\mathbf{x}, z, 1) \qquad \mathrm{CARRY}_{s,\ell}(\mathbf{x}, \overline{z}, 1). \tag{19}$$

For every $p \in \{0, \ldots, B\}$ define formulas

$$\begin{aligned} R(p) &\equiv S_\ell(\mathbf{x}, \mathbf{y}, 1; p) = S_\ell(\mathbf{x}, z, 1; p) + S_\ell(\mathbf{y}, \overline{z}, 1; p) + 1, \\ S(p) &\equiv S_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq S_\ell(\mathbf{y}, \overline{z}, 1; p) + 1. \\ T(p) &\equiv S_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq S_\ell(\mathbf{x}, z, 1; p) + 1. \end{aligned}$$

For the sake of argument, let $M_1$ refer to the automaton on input $\mathbf{x}, z, 1$, let $M_2$ refer to the automaton on input $\mathbf{y}, \overline{z}, 1$, and let $M_3$ refer to the automaton on input $\mathbf{x}, \mathbf{y}, 1$. Intuitively, what we want to show is that, for every $p \in \{1, \ldots, B\}$, if neither $M_1$ nor $M_2$ have accepted yet by time $p$, then the states of $M_1$, $M_2$ and $M_3$ at time $p$ stay related as in $R(p)$. On the other hand, if $M_1$ has already accepted by time $p$ but $M_2$ has not, then the states of $M_1$, $M_2$ and $M_3$ at time $p$ stay related as in $S(p)$. Similarly, if $M_1$ has not yet accepted by time $p$ but $M_1$ has, then the states of $M_1$, $M_2$ and $M_3$ at time $p$ stay related as in $T(p)$. This will guarantee that by the time both $M_1$ and $M_2$ have accepted, $M_3$ will have accepted as well since its state is always *ahead*.

We will prove these facts by induction on $p$. For later reference we state the base case and the inductive cases of induction as claims. The first claim states that all three automata start at the initial state.

**Claim 2.** $S_\ell(\mathbf{x}, \mathbf{y}, 1; 0) = S_\ell(\mathbf{x}, z, 1; 0) + S_\ell(\mathbf{y}, \overline{z}, 1; 0) + 1.$

*Proof.* This is immediate from the fact that $S(\mathbf{x}, \mathbf{y}, 1; 0) = S(\mathbf{x}, z, 1; 0) = S(\mathbf{y}, \overline{z}, 1; 0) = -1$. □

The second claim states that if neither $M_1$ nor $M_2$ have accepted by time $p$ and the relationship $R(p-1)$ holds, then either $M_3$ accepts by time $p$ or the relationship $R(p)$ still holds.

**Claim 3.** *Let* $1 \leq p \leq B - 1$.
*If both*

1. $-r < N_\ell(\mathbf{x}, z, 1; p) < 0 \wedge -s < N_\ell(\mathbf{y}, \overline{z}, 1; p) < 0$, *and*
2. $S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) = S_\ell(\mathbf{x}, z, 1; p-1) + S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1$,

*then either*

1. $N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq 0$, *or*
2. $-k < N_\ell(\mathbf{x}, \mathbf{y}, 1; p) < 0 \wedge S_\ell(\mathbf{x}, \mathbf{y}, 1; p) = S_\ell(\mathbf{x}, z, 1; p) + S_\ell(\mathbf{y}, \overline{z}, 1; p) + 1$.

*Proof.* If $N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq 0$ there is nothing to prove. Assume then $N_\ell(\mathbf{x}, \mathbf{y}, 1; p) < 0$. The assumptions $-r < N_\ell(\mathbf{x}, z, 1; p) < 0$ and $-s < N_\ell(\mathbf{y}, \overline{z}, 1; p) < 0$ together with Lemma 6 give $S_\ell(\mathbf{x}, z, 1; p) = N_\ell(\mathbf{x}, z, 1; p)$ and $S_\ell(\mathbf{y}, \overline{z}, 1; p) = N_\ell(\mathbf{y}, \overline{z}, 1; p)$. Since $p \leq B - 1$ we have

$$\begin{aligned} N_\ell(\mathbf{x}, \mathbf{y}, 1; p) &= 2S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) + (\mathbf{x}_r + \mathbf{y}_r) \\ &= 2(S_\ell(\mathbf{x}, z, 1; p-1) + S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1) + (\mathbf{x}_p + \mathbf{y}_p) \\ &= 2S_\ell(\mathbf{x}, z, 1; p-1) + (\mathbf{x}_p + z_p) + 2S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + (\mathbf{y}_p + 1 - z_p) + 1 \\ &= N_\ell(\mathbf{x}, z, 1; p) + N_\ell(\mathbf{y}, \overline{z}, 1; p) + 1, \end{aligned}$$

where the second equality follows from the assumption $S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) = S_\ell(\mathbf{x}, z, 1; p-1) + S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1$. From the assumptions that $-r < N_\ell(\mathbf{x}, z, 1; p) < 0$ and $-s < N_\ell(\mathbf{y}, \overline{z}, 1; p) < 0$ we conclude that

$$N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq (1-r) + (1-s) + 1 > -k.$$

At this point we have $-k < N_\ell(\mathbf{x}, \mathbf{y}, 1; p) < 0$ and we can apply Lemma 6 to obtain $S_\ell(\mathbf{x}, \mathbf{y}, 1; p) = N_\ell(\mathbf{x}, \mathbf{y}, 1; p)$. Putting all these together we get $S_\ell(\mathbf{x}, \mathbf{y}, 1; p) = S_\ell(\mathbf{x}, z, 1; p) + S_\ell(\mathbf{y}, \overline{z}, 1; p) + 1$. □

The third claim states that if $M_1$ accepts at time $p$ but $M_2$ has not accepted yet by time $p$ and moreover the relationship $R(p-1)$ holds, then either $M_3$ accepts by time $p$ or the relationship $S(p)$ starts to hold.

**Claim 4.** *Let* $1 \leq p \leq B-1$.
*If both*

1. $N_\ell(\mathbf{x}, z, 1; p) \geq 0 \wedge -s < N_\ell(\mathbf{y}, \overline{z}, 1; p) < 0$, *and*
2. $S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) = S_\ell(\mathbf{x}, z, 1; p-1) + S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1$,

*then either*

1. $N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq 0$, *or*
2. $-k < N_\ell(\mathbf{x}, \mathbf{y}, 1; p) < 0 \wedge S_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq S_\ell(\mathbf{y}, \overline{z}, 1; p) + 1$.

*Proof.* If $N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq 0$ there is nothing to prove. Assume then $N_\ell(\mathbf{x}, \mathbf{y}, 1; p) < 0$. The assumption $-s < N_\ell(\mathbf{y}, \overline{z}, 1; p) < 0$ together with Lemma 6 gives $S_\ell(\mathbf{y}, \overline{z}, 1; p) = N_\ell(\mathbf{y}, \overline{z}, 1; p)$. Since $p \leq B-1$ we have

$$\begin{aligned}
N_\ell(\mathbf{x}, \mathbf{y}, 1; p) &= 2S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) + (\mathbf{x}_p + \mathbf{y}_p) \\
&= 2(S_\ell(\mathbf{x}, z, 1; p-1) + S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1) + (\mathbf{x}_p + \mathbf{y}_p) \\
&= 2S_\ell(\mathbf{x}, z, 1; p-1) + (\mathbf{x}_p + z_p) + 2S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + (\mathbf{y}_p + 1 - z_p) + 1 \\
&= N_\ell(\mathbf{x}, z, 1; p) + N_\ell(\mathbf{y}, \overline{z}, 1; p) + 1 \\
&\geq N_\ell(\mathbf{y}, \overline{z}, 1; p) + 1,
\end{aligned}$$

where the second equality follows from the assumption $S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) = S_\ell(\mathbf{x}, z, 1; p-1) + S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1$, and the inequality follows from the assumption $N_\ell(\mathbf{x}, z, 1; p) \geq 0$. From the assumption $-s < N_\ell(\mathbf{y}, \overline{z}, 1; p) < 0$ we conclude that

$$N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq 1 - s + 1 > -k.$$

At this point we have $-k < N_\ell(\mathbf{x}, \mathbf{y}, 1; p) < 0$ and we can apply Lemma 6 to obtain $S_\ell(\mathbf{x}, \mathbf{y}, 1; p) = N_\ell(\mathbf{x}, \mathbf{y}, 1; p)$. Putting all these together we get $S_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq S_\ell(\mathbf{y}, \overline{z}, 1; p) + 1$. □

A claim symmetric to the above would state the symmetric property that when $M_1$ has not accepted yet by time $p$ and $M_2$ accepts at time $p$ and moreover the relationship $R(p-1)$ holds, then the relationshop $T(p)$ starts to hold.

The forth claim states that if $M_2$ has not accepted yet by time $p$ and the relationship $S(p-1)$ holds, then either $M_3$ accepts at time $p$ or the relationship $S(p)$ still holds.

**Claim 5.** *Let $1 \leq p \leq B-1$.*
*If both*

1. *$-s < N_\ell(\mathbf{y}, \overline{z}, 1; p) < 0$, and*
2. *$S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) \geq S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1$,*

*then either*

1. *$N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq 0$, or*
2. *$-k < N_\ell(\mathbf{x}, \mathbf{y}, 1; p) < 0 \wedge S_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq S_\ell(\mathbf{y}, \overline{z}, 1; p) + 1$.*

*Proof of Lemma 5.* If $N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq 0$ there is nothing to prove. Assume then $N_\ell(\mathbf{x}, \mathbf{y}, 1; p) < 0$. The assumption $-s < N_\ell(\mathbf{y}, \overline{z}, 1; p) < 0$ together with Lemma 6 gives $S_\ell(\mathbf{y}, \overline{z}, 1; p) = N_\ell(\mathbf{y}, \overline{z}, 1; p)$. Since $p \leq B-1$ we have

$$\begin{aligned}
N_\ell(\mathbf{x}, \mathbf{y}, 1; p) &= 2S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) + (\mathbf{x}_p + \mathbf{y}_p) \\
&\geq 2(S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1) + (\mathbf{x}_p + \mathbf{y}_p) \\
&= 2S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + (\mathbf{y}_p + 1 - z_p) + (\mathbf{x}_p + z_p) + 1 \\
&= N_\ell(\mathbf{y}, \overline{z}, 1; p) + (\mathbf{x}_p + z_p) + 1 \\
&\geq N_\ell(\mathbf{y}, \overline{z}, 1; p) + 1,
\end{aligned}$$

where the first inequality follows from the assumption $S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) \geq S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1$, and the second inequality follows from the fact that $\mathbf{x}_p + z_p \geq 0$. From the assumption $-s < N_\ell(\mathbf{y}, \overline{z}, 1; p) < 0$ we conclude that

$$N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq 1 - s + 1 > -k.$$

At this point we have $-k < N_\ell(\mathbf{x}, \mathbf{y}, 1; p) < 0$ and we can apply Lemma 6 to obtain $S_\ell(\mathbf{x}, \mathbf{y}, 1; p) = N_\ell(\mathbf{x}, \mathbf{y}, 1; p)$. Putting all these together we get $S_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq S_\ell(\mathbf{y}, \overline{z}, 1; p) + 1$. □

A claim symmetric to the above would state the symmetric property that if $M_1$ has not accepted yet and the relationship $T(p-1)$ holds, then the relationship $T(p)$ still holds.

The fifth claim states that if both $M_1$ and $M_2$ accept at time $p$ and the relationship $R(p-1)$ holds, then $M_3$ also accepts at time $p$.

**Claim 6.** *Let $1 \leq p \leq B$.*
*If both*

1. *$N_\ell(\mathbf{x}, z, 1; p) \geq 0 \wedge N_\ell(\mathbf{y}, \overline{z}, 1; p) \geq 0$, and*
2. *$S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) = S_\ell(\mathbf{x}, z, 1; p-1) + S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1$,*

*then*

$N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq 0.$

*Proof.* We distinguish the cases $p \leq B-1$ and $p = B$. In case $p \leq B-1$ we have

$$\begin{aligned} N_\ell(\mathbf{x}, \mathbf{y}, 1; p) &= 2S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) + (\mathbf{x}_p + \mathbf{y}_p) \\ &= 2(S_\ell(\mathbf{x}, z, 1; p-1) + S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1) + (\mathbf{x}_p + \mathbf{y}_p) \\ &= 2S_\ell(\mathbf{x}, z, 1; p-1) + (\mathbf{x}_p + z_p) + 2S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + (\mathbf{y}_p + 1 - z_p) + 1 \\ &= N_\ell(\mathbf{x}, z, 1; p) + N_\ell(\mathbf{y}, \overline{z}, 1; p) + 1, \end{aligned}$$

where the second equality follows from the assumption $S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) = S_\ell(\mathbf{x}, z, 1; p-1) + S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1$. From the assumptions $N_\ell(\mathbf{x}, z, 1; p) \geq 0$ and $N_\ell(\mathbf{y}, \overline{z}, 1; p) \geq 0$ we conclude that $N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq 1 \geq 0$. The case $p = B$ is similar: we have

$$\begin{aligned} N_\ell(\mathbf{x}, \mathbf{y}, 1; B) &= 2S_\ell(\mathbf{x}, \mathbf{y}, 1; B-1) + (\mathbf{x}_B + \mathbf{y}_B + 1) \\ &= 2(S_\ell(\mathbf{x}, z, 1; B-1) + S_\ell(\mathbf{y}, \overline{z}, 1; B-1) + 1) + (\mathbf{x}_B + \mathbf{y}_B + 1) \\ &= 2S_\ell(\mathbf{x}, z, 1; B-1) + (\mathbf{x}_B + z_B + 1) + 2S_\ell(\mathbf{y}, \overline{z}, 1; B-1) + (\mathbf{y}_B + 1 - z_B + 1) \\ &= N_\ell(\mathbf{x}, z, 1; B) + N_\ell(\mathbf{y}, \overline{z}, 1; B) \end{aligned}$$

where the second equality follows from the assumption $S_\ell(\mathbf{x}, \mathbf{y}, 1; B-1) = S_\ell(\mathbf{x}, z, 1; B-1) + S_\ell(\mathbf{y}, \overline{z}, 1; B-1) + 1$. From the assumptions $N_\ell(\mathbf{x}, z, 1; B) \geq 0$ and $N_\ell(\mathbf{y}, \overline{z}, 1; B) \geq 0$ we conclude that $N_\ell(\mathbf{x}, \mathbf{y}, 1; B) \geq 0$. □

The sixth claim states that if $M_2$ accepts at time $p$ and the relationship $S(p-1)$ holds, then $M_3$ also accepts at time $p$.

**Claim 7.** *Let* $1 \leq p \leq B$.
*If both*

1. $N_\ell(\mathbf{y}, \overline{z}, 1; p) \geq 0$, *and*
2. $S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) \geq S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1$,

*then*

$$N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq 0.$$

*Proof.* We distinguish the cases $p \leq B-1$ and $p = B$. In case $p \leq B-1$ we have

$$\begin{aligned} N_\ell(\mathbf{x}, \mathbf{y}, 1; p) &= 2S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) + (\mathbf{x}_p + \mathbf{y}_p) \\ &\geq 2(S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1) + (\mathbf{x}_p + \mathbf{y}_p) \\ &= 2S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + (\mathbf{y}_p + 1 - z_p) + (\mathbf{x}_p + z_p) + 1 \\ &= N_\ell(\mathbf{y}, \overline{z}, 1; p) + (\mathbf{x}_r + z_r) + 1 \\ &\geq N_\ell(\mathbf{y}, \overline{z}, 1; p) + 1, \end{aligned}$$

where the first inequality follows from the assumption $S_\ell(\mathbf{x}, \mathbf{y}, 1; p-1) \geq S_\ell(\mathbf{y}, \overline{z}, 1; p-1) + 1$, and the second inequality follows from the fact that $\mathbf{x}_p + z_p \geq 0$. From the assumption $N_\ell(\mathbf{y}, \overline{z}, 1; p) \geq 0$ we conclude that $N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq 1 \geq 0$. The case $p = B$ is similar: we have

$$\begin{aligned} N_\ell(\mathbf{x}, \mathbf{y}, 1; B) &= 2S_\ell(\mathbf{x}, \mathbf{y}, 1; B-1) + (\mathbf{x}_B + \mathbf{y}_B + 1) \\ &\geq 2(S_\ell(\mathbf{y}, \overline{z}, 1; B-1) + 1) + (\mathbf{x}_B + \mathbf{y}_B + 1) \\ &= 2S_\ell(\mathbf{y}, \overline{z}, 1; B-1) + (\mathbf{y}_B + 1 - z_B + 1) + (\mathbf{x}_B + z_B) \\ &= N_\ell(\mathbf{y}, \overline{z}, 1; B) + (\mathbf{x}_B + z_B) \\ &\geq N_\ell(\mathbf{y}, \overline{z}, 1; B), \end{aligned}$$

where the first inequality follows from the assumption $S_\ell(\mathbf{x}, \mathbf{y}, 1; B-1) \geq S_\ell(\mathbf{y}, \overline{z}, 1; B-1)+1$, and the second inequality follows from the fact that $\mathbf{x}_B + z_B \geq 0$. From the assumption $N_\ell(\mathbf{y}, \overline{z}, 1; B) \geq 0$ we conclude that $N_\ell(\mathbf{x}, \mathbf{y}, 1; B) \geq 0$. □

A claim symmetric to the above would state the symmetric property that if $M_1$ accepts at time $p$ and the relationship $T(p-1)$ holds, then $M_3$ also accepts at time $p$.

Next we turn to the formal proof. For every $p \in \{1, \ldots, B\}$ define formulas

$$\begin{array}{rcl}
F^+(p) & \equiv & N_\ell(\mathbf{x}, z, 1; p) \geq 0, \\
G^+(p) & \equiv & N_\ell(\mathbf{y}, \overline{z}, 1; p) \geq 0, \\
H^+(p) & \equiv & N_\ell(\mathbf{x}, \mathbf{y}, 1; p) \geq 0, \\
F^-(p) & \equiv & -r < N_\ell(\mathbf{x}, z, 1; p) < 0, \\
G^-(p) & \equiv & -s < N_\ell(\mathbf{y}, \overline{z}, 1; p) < 0, \\
H^-(p) & \equiv & -k < N_\ell(\mathbf{x}, \mathbf{y}, 1; p) < 0.
\end{array}$$

Claims 2, 3, 4, 5, 6 and 7 state that all the base-case, inductive-case and terminating-case formulas in the scheme of induction Lemma 9 below are tautologies. Since these are constant-size $\Sigma_{1,\mathsf{c}}$-formulas, by completeness they have $\Sigma_{1,\mathsf{c}}$-Frege proofs of constant size. Lemma 8.1 now follow from this scheme applied to these formulas and the hypothesis formulas in (19).

**Lemma 9.** *The following assertion has polynomial-size $\Sigma_{2,\mathsf{c}}$-Frege proofs:*
*Given the* base-case*:*

$R(0)$,

*the* inductive-case *for every $p \in \{1, \ldots, B\}$:*

1. $\neg F^-(p) \vee \neg G^-(p) \vee \neg R(p-1) \vee H^+(p) \vee H^-(p)$,
2. $\neg F^-(p) \vee \neg G^-(p) \vee \neg R(p-1) \vee H^+(p) \vee R(p)$,
3. $\neg F^+(p) \vee \neg G^-(p) \vee \neg R(p-1) \vee H^+(p) \vee H^-(p)$,
4. $\neg F^+(p) \vee \neg G^-(p) \vee \neg R(p-1) \vee H^+(p) \vee S(p)$,
5. $\neg F^-(p) \vee \neg G^+(p) \vee \neg R(p-1) \vee H^+(p) \vee H^-(p)$,
6. $\neg F^-(p) \vee \neg G^+(p) \vee \neg R(p-1) \vee H^+(p) \vee T(p)$,
7. $\neg G^-(p) \vee \neg S(p-1) \vee H^+(p) \vee H^-(p)$,
8. $\neg G^-(p) \vee \neg S(p-1) \vee H^+(p) \vee S(p)$,
9. $\neg F^-(p) \vee \neg T(p-1) \vee H^+(p) \vee H^-(p)$,
10. $\neg F^-(p) \vee \neg T(p-1) \vee H^+(p) \vee T(p)$,

*the* terminating-case *for every $p \in \{1, \ldots, B\}$:*

1. $\neg F^+(p) \vee \neg G^+(p) \vee \neg R(p-1) \vee H^+(p)$,
2. $\neg G^+(p) \vee \neg S(p-1) \vee H^+(p)$,
3. $\neg F^+(p) \vee \neg T(p-1) \vee H^+(p)$,

*and the* hypothesis*:*

*1.* $(\exists p : 1 \leq p \leq B)(F^+(p) \wedge (\forall q : 1 \leq q \leq p-1)(F^-(q)))$,

*2.* $(\exists p : 1 \leq p \leq B)(G^+(p) \wedge (\forall q : 1 \leq q \leq p-1)(G^-(q)))$,

*conclude:*

$$(\exists p : 1 \leq p \leq B)(H^+(p) \wedge (\forall q : 1 \leq q \leq p-1)(H^-(q))).$$

*Proof.* We start fixing some notation. For $i \in \{1, \ldots, 10\}$ and $p \in \{1, \ldots, B\}$, let (I.$i$.$p$) denote the inductive-case formula numbered $i$ in the list, for the indicated value of $p$. Similarly, for $i \in \{1, \ldots, 3\}$ and $p \in \{1, \ldots, B\}$, let (T.$i$.$p$) denote the terminating-case formula numbered $i$ in the list, for the indicated value of $p$. Let $F^*(p)$ denote the formula

$$(\forall q : 1 \leq q < p)(F^-(q)),$$

and similarly for $G^*(p)$ and $H^*(p)$.

For every pair $a, b \in \{1, \ldots, B\}$ we will give a proof of the following assertion:

$$\frac{F^+(a) \wedge F^*(a) \qquad\qquad G^+(b) \wedge G^*(b)}{(\exists p : 1 \leq p \leq B)(H^+(p) \wedge H^*(p))}. \tag{20}$$

The result will follow from the proof-scheme of pairwise case-analysis Lemma 3.

The proof splits into several cases, many of which are symmetric versions of some other: case $a = b = 1$, case $a > b = 1$, case $b > a = 1$, case $a = b > 1$, case $a > b > 1$, and case $b > a > 1$. Since all proofs follow a common pattern we give the details for the last case only. Assume from now on that $b > a > 1$. We start showing how to get, for every $q \in \{1, \ldots, a-1\}$, the formulas

$$\neg F^*(q+1) \vee \neg G^*(q+1) \vee (\exists p : 1 \leq p \leq q)(H^+(p) \wedge H^*(p)) \vee H^-(q) \tag{21}$$

$$\neg F^*(q+1) \vee \neg G^*(q+1) \vee (\exists p : 1 \leq p \leq q)(H^+(p) \wedge H^*(p)) \vee R(q). \tag{22}$$

For $q = 1$ we apply CUT between the base-case formula $R(0)$ and (I.1.1) to get (21.1). Similarly, apply CUT between the base-case formula $R(0)$ and (I.2.1) to get (22.1). For $2 \leq q \leq a-1$, and assuming we have (21.$r$) and (22.$r$) for every $r \in \{1, \ldots, q-1\}$, first we apply CUT between (22.$q-1$) and (I.1.$q$), and between (22.$q-1$) and (I.2.$q$). These give

$$\neg F^*(q) \vee \neg G^*(q) \vee \neg F^-(q) \vee \neg G^-(q) \vee (\exists p : 1 \leq p < q)(H^+(p) \wedge H^*(p)) \vee H^+(q) \vee H^-(q) \tag{23}$$

$$\neg F^*(q) \vee \neg G^*(q) \vee \neg F^-(q) \vee \neg G^-(q) \vee (\exists p : 1 \leq p < q)(H^+(p) \wedge H^*(p)) \vee H^+(q) \vee R(q). \tag{24}$$

Then we apply IOC on (21.1), $\ldots$, (21.$q-1$) and (23.$q$) to get (21.$q$). Similarly, we apply IOC on (21.1), $\ldots$, (21.$q-1$) and (24.$q$) gives (22.$q$). We continue with CUTs between (22.$a-1$) and (I.3.$a$), and between (22.$a-1$) and (I.4.$a$), to get

$$\neg F^*(a) \vee \neg G^*(a) \vee \neg F^+(a) \vee \neg G^-(a) \vee (\exists p : 1 \leq p < a)(H^+(p) \wedge H^*(p)) \vee H^+(a) \vee H^-(a) \tag{25}$$

$$\neg F^*(a) \vee \neg G^*(a) \vee \neg F^+(a) \vee \neg G^-(a) \vee (\exists p : 1 \leq p < a)(H^+(p) \wedge H^*(p)) \vee H^+(a) \vee S(a). \tag{26}$$

Then IOC on (21.1), $\ldots$, (21.$a-1$) and (25/26.$a$) gives

$$\neg F^+(a) \vee \neg F^*(a) \vee \neg G^*(a+1) \vee (\exists p : 1 \leq p \leq a)(H^+(p) \wedge H^*(p)) \vee H^-(a) \tag{27}$$

$$\neg F^+(a) \vee \neg F^*(a) \vee \neg G^*(a+1) \vee (\exists p : 1 \leq p \leq a)(H^+(p) \wedge H^*(p)) \vee S(a). \tag{28}$$

Next we get, for every $q \in \{a+1, \ldots, b-1\}$, the formulas

$$\neg F^+(a) \vee \neg F^*(a) \vee \neg G^*(q+1) \vee (\exists p : 1 \leq p \leq q)(H^+(p) \wedge H^*(p)) \vee H^-(q) \tag{29}$$
$$\neg F^+(a) \vee \neg F^*(a) \vee \neg G^*(q+1) \vee (\exists p : 1 \leq p \leq q)(H^+(p) \wedge H^*(p)) \vee S(q). \tag{30}$$

To achieve this we use (I.7.$q$) and (I.8.$q$) for $q \in \{a+1, \ldots, b-1\}$ in a similar fashion as above. At this point we are almost ready to conclude. Apply CUT on (30.$b-1$) and (T.2.$b$) to get

$$\neg F^+(a) \vee \neg F^*(a) \vee \neg G^*(b) \vee \neg G^+(b) \vee (\exists p : 1 \leq p < b)(H^+(p) \wedge H^*(p)) \vee H^+(b). \tag{31}$$

Now we apply IOC on (21.1), ..., (21.$a-1$), (27.$a$), (29.$a+1$), ..., (29.$b-1$) and (31) to get

$$\neg F^+(a) \vee \neg F^*(a) \vee \neg G^+(b) \vee \neg G^*(b) \vee (\exists p : 1 \leq p \leq B)(H^+(p) \wedge H^*(p)).$$

Finally apply CUT between this an the first hypothesis in (20), and CUT between the result and the second hypothesis in (20). This completes the proof. □

**Proof of Lemma 8.2** For every $p \in \{0, \ldots, B\}$, let $d_p \in \{0, 1\}$ be the *bit of carry* at position $p$ while adding the numbers represented by $a_1 \ldots a_B$ and $b_1 \ldots b_B$ in binary notation. It will be useful to keep in mind that $d_B = 0$ and

$$\begin{array}{lcl} d_{p-1} & = & (a_p \wedge b_p) \vee (a_p \wedge d_p) \vee (b_p \wedge d_p) \\ c_p & = & a_p \oplus b_p \oplus d_p \end{array}$$

for every $p \in \{1, \ldots, B\}$.

For every $p \in \{0, \ldots, B\}$ we define three bit-strings

$$\begin{array}{lcllllll} \alpha_p & = & a_1 & \ldots & a_{p-1} & a_p & c_{p+1} & \ldots & c_B, \\ \beta_p & = & b_1 & \ldots & b_{p-1} & b_p & 0 & \ldots & 0, \\ \delta_p & = & 0 & \ldots & 0 & d_p & 0 & \ldots & 0. \end{array}$$

Note that $\delta_0$, $\delta_B$ and $\beta_0$ are all three the all-zero string. Note also that $\alpha_B = a$, $\beta_B = b$, and $\alpha_0 = c$. Hence the following assertions are valid:

$$\frac{\mathrm{CARRY}_{k,\ell}(\mathbf{x}, a, b)}{\mathrm{CARRY}_{k,\ell}(\mathbf{x}, \alpha_B, \beta_B, \delta_B)} \qquad \frac{\mathrm{CARRY}_{k,\ell}(\mathbf{x}, \alpha_0, \beta_0, \delta_0)}{\mathrm{CARRY}_{k,\ell}(\mathbf{x}, c)}. \tag{32}$$

Therefore it will suffice to give small $\Sigma_2$-proofs of these and, for every $p \in \{1, \ldots, B\}$, of

$$\frac{\mathrm{CARRY}_{k,\ell}(\mathbf{x}, \alpha_p, \beta_p, \delta_p)}{\mathrm{CARRY}_{k,\ell}(\mathbf{x}, \alpha_{p-1}, \beta_{p-1}, \delta_{p-1})}. \tag{33}$$

The result will follow by chaining all of them together.

The small $\Sigma_2$-proofs of (32) are direct instances of Lemma 7.2. Let us focus on (33.$p$) for a fixed $p \in \{1, \ldots, B\}$. Reserve notation:

$$\begin{array}{lcl} \mathbf{z} & = & (\mathbf{x}, \alpha_p, \beta_p, \delta_p), \\ \mathbf{z}' & = & (\mathbf{x}, \alpha_{p-1}, \beta_{p-1}, \delta_{p-1}). \end{array}$$

Define formulas

$$\begin{array}{rcl} F^+(r) & \equiv & N_\ell(\mathbf{z}; r) \geq 0, \\ G^+(r) & \equiv & N_\ell(\mathbf{z}'; r) \geq 0, \\ F^-(r) & \equiv & -k < N_\ell(\mathbf{z}; r) < 0, \\ G^-(r) & \equiv & -k < N_\ell(\mathbf{z}'; r) < 0, \end{array}$$

for every $r \in \{1, \ldots, B\}$, and formulas

$$\begin{array}{rcl} F^*(q) & \equiv & (\forall r : 0 < r < q)(F^-(r)), \\ G^*(q) & \equiv & (\forall r : 0 < r < q)(G^-(r)), \end{array}$$

for every $q \in \{1, \ldots, B\}$. The goal (33.$p$) will follow from the proof-scheme of case analysis Lemma 4 if we succeed in proving

$$\frac{F^+(q) \wedge F^*(q)}{\mathrm{CARRY}_{k,\ell}(\mathbf{z}')} \tag{34}$$

for every $q \in \{1, \ldots, B\}$. In order to prove (34.$q$) for a fixed $q \in \{1, \ldots, B\}$ we distinguish by cases according to the value of $d_{p-1}$.

Case $d_{p-1} = 0$: In case $d_{p-1} = 0$ at most one among $a_p$, $b_p$ and $d_p$ is 1. Since $c_p = a_p \oplus b_p \oplus d_p$, this means that the following identity holds:

$$c_p = a_p + b_p + d_p. \tag{35}$$

The fact that $\mathbf{z}$ and $\mathbf{z}'$ differ only in positions $p$ and $p - 1$, together with identity (35) and the assumption $d_{p-1} = 0$, shows that for every $r \in \{1, \ldots, B\}$ we have

$$A_\ell(\mathbf{z}'; r) = A_\ell(\mathbf{z}; r), \tag{36}$$

$$S_\ell(\mathbf{z}'; r) = S_\ell(\mathbf{z}; r). \tag{37}$$

We use these facts to argue that for every $r \in \{1, \ldots, B\}$ also

$$N_\ell(\mathbf{z}'; r) = N_\ell(\mathbf{z}; r). \tag{38}$$

The case $r = p$ follows from the following derivation:

$$\begin{aligned} N_\ell(\mathbf{z}'; p) &= 2S_\ell(\mathbf{z}'; p - 1) + c_p \\ &= 2S_\ell(\mathbf{z}; p - 1) + c_p \\ &= 2S_\ell(\mathbf{z}; p - 1) + a_p + b_p + d_p \\ &= N_\ell(\mathbf{z}; p), \end{aligned}$$

where the second equality follows from (37), and the third equality follows from (35). The case $r \neq p$ follows again from (37) and the assumption $d_{p-1} = 0$ in the special case $r = p - 1$.

The validity of equation (38) implies that, for every $r \in \{1, \ldots, B\}$, the following are tautologies

$$\neg F^+(r) \vee G^+(r), \tag{39}$$

$$\neg F^-(r) \vee G^-(r). \tag{40}$$

These are constant-size $\Sigma_{1,\mathsf{c}}$-formulas and therefore, by completeness, they have $\Sigma_{1,\mathsf{c}}$-Frege proofs of constant size. From these, the goal in (34.$q$) is obtained by applying IOC on (40.1), $\ldots$, (40.$q - 1$) and (39.$q$), and then applying CUT with hypothesis.

Case $d_{p-1} = 1$: In case $d_{p-1} = 1$ at least two among $a_p$, $b_p$ and $d_p$ are 1. Since $c_p = a_p \oplus b_p \oplus d_p$, this means that the following identity holds:

$$c_p = a_p + b_p + d_p - 2. \tag{41}$$

In this case equation (36) is no longer guaranteed for every $r \in \{1, \ldots, B\}$. However, we can argue that for every $r \in \{1, \ldots, B\}$ we have the following:

$$\begin{aligned}
&A_\ell(\mathbf{z}'; r) = A_\ell(\mathbf{z}; r) && \text{if } r \neq p-1 \text{ and } r \neq p+\ell, \\
&A_\ell(\mathbf{z}'; r) = A_\ell(\mathbf{z}; r) + 1 && \text{if } r = p-1, \\
&A_\ell(\mathbf{z}'; r) = A_\ell(\mathbf{z}; r) - 2 \cdot 2^{\ell-1} && \text{if } r = p+\ell.
\end{aligned}$$

The case $r = p-1$ follows directly from the assumption $d_{p-1} = 1$. The case $r = p+\ell$ follows from (41). The case $p-1 < r < p+\ell$ uses both the assumption $d_{p-1} = 1$ and identity (41). All remaining cases where either $r < p-1$ or $r > p+\ell$ are trivial since $\mathbf{z}$ and $\mathbf{z}'$ differ only in positions $p$ and $p-1$.

Let $\ell' = \min\{\ell, p-1\}$. Taking mod $2^{\ell'}$ the above implies that for every $r \in \{1, \ldots, B\}$ we have the following:

$$\begin{aligned}
&S_\ell(\mathbf{z}'; r) = S_\ell(\mathbf{z}; r) && \text{if } r \neq p-1, && (42)\\
&S_\ell(\mathbf{z}'; r) = S_\ell(\mathbf{z}; r) + 1 && \text{if } r = p-1 \text{ and } A_\ell(\mathbf{z}; p-1) \not\equiv -1 \mod 2^{\ell'}, && (43)\\
&S_\ell(\mathbf{z}'; r) = -2^{\ell'} \text{ and } S_\ell(\mathbf{z}; r) = -1 && \text{if } r = p-1 \text{ and } A_\ell(\mathbf{z}; p-1) \equiv -1 \mod 2^{\ell'}. && (44)
\end{aligned}$$

Next we argue that

$$\begin{aligned}
&N_\ell(\mathbf{z}'; r) = N_\ell(\mathbf{z}; r) && \text{if } r \neq p-1 \text{ and } r \neq p, && (45)\\
&N_\ell(\mathbf{z}'; r) = N_\ell(\mathbf{z}; r) + 1 && \text{if } r = p-1, && (46)\\
&N_\ell(\mathbf{z}'; r) = N_\ell(\mathbf{z}; r) && \text{if } r = p \text{ and } A_\ell(\mathbf{z}; p-1) \not\equiv -1 \mod 2^{\ell'}. && (47)
\end{aligned}$$

The case where $r \neq p-1$ and $r \neq p$ follows from (42) and the fact that $\mathbf{z}$ and $\mathbf{z}'$ differ only in positions $p$ and $p-1$. The case $r = p-1$ follows from the following derivation:

$$\begin{aligned}
N_\ell(\mathbf{z}'; p-1) &= 2S_\ell(\mathbf{z}'; p-2) + \mathbf{x}_{p-1} + a_{p-1} + b_{p-1} + d_{p-1} \\
&= 2S_\ell(\mathbf{z}; p-2) + (\mathbf{x}_{p-1} + a_{p-1} + b_{p-1}) + 1 \\
&= N_\ell(\mathbf{z}; p-1) + 1.
\end{aligned}$$

The case where $r = p$ and $A_\ell(\mathbf{z}; p-1) \not\equiv -1 \mod 2^{\ell'}$ follows from the following derivation:

$$\begin{aligned}
N_\ell(\mathbf{z}'; p) &= 2S_\ell(\mathbf{z}'; p-1) + \mathbf{x}_p + c_p && (48)\\
&= 2S_\ell(\mathbf{z}; p-1) + \mathbf{x}_p + c_p + 2 && (49)\\
&= 2S_\ell(\mathbf{z}; p-1) + \mathbf{x}_p + a_p + b_p + d_p && (50)\\
&= N_\ell(\mathbf{z}; p), && (51)
\end{aligned}$$

where (49) follows from (43), and (50) follows from identity (41).

At this point we are ready to complete the proof. We start defining one more formula:

$$C \quad \equiv \quad (A_\ell(\mathbf{z}; p-1) \equiv -1 \mod 2^{\ell'}).$$

Assuming $-k < N_\ell(\mathbf{z}; p-1) < 0$ we have $N_\ell(\mathbf{z}; p-1) = S_\ell(\mathbf{z}; p-1)$ by Lemma 6. Under the further assumption that $C$ holds we get $S_\ell(\mathbf{z}; p-1) = -1$ from (44). Under these conditions (46) gives

$$\begin{aligned} N_\ell(\mathbf{z}'; p-1) &= N_\ell(\mathbf{z}; p-1) + 1 \\ &= S_\ell(\mathbf{z}; p-1) + 1 \\ &= 0. \end{aligned}$$

On the other hand, under the assumption that $C$ does not hold, equation (47) gives

$$N_\ell(\mathbf{z}'; p) = N_\ell(\mathbf{z}; p).$$

Together with equations (45) and (46), this reasoning establishes that the following are tautologies for every $r \in \{1, \ldots, B\}$:

$$\begin{aligned}
&\neg F^+(r) \vee G^+(r) && \text{if } r \neq p-1 \text{ and } r \neq p, && (52)\\
&\neg F^-(r) \vee G^-(r) && \text{if } r \neq p-1 \text{ and } r \neq p, && (53)\\
&\neg F^+(r) \vee G^+(r) && \text{if } r = p-1, && (54)\\
&\neg F^-(r) \vee G^+(r) \vee G^-(r) && \text{if } r = p-1, && (55)\\
&\neg F^-(r) \vee \neg C \vee G^+(r) && \text{if } r = p-1, && (56)\\
&\neg F^+(r) \vee C \vee G^+(r) && \text{if } r = p, && (57)\\
&\neg F^-(r) \vee C \vee G^-(r) && \text{if } r = p. && (58)
\end{aligned}$$

All these are constant-size $\Sigma_{1,\mathbf{c}}$-formulas and hence, by completeness, they have $\Sigma_{1,\mathbf{c}}$-Frege proofs of constant size. If $q \leq p-2$ we can put together the goal in (34.$q$) using only (53.1), …, (53.$q-1$), and (52.$q$). If $q = p-1$ we can put together the goal (34.$q$) using only (53.1), …, (53.$q-1$), and (54). If $q = p$ we apply CUT between (56) and (57) to get

$$\neg F^+(p) \vee \neg F^-(p-1) \vee G^+(p-1) \vee G^+(p),$$

and then use this together with (53.1), …, (53.$p-2$), and (55) to work-out the goal in (34.$q$). Finally, if $q \geq p+1$ we apply CUT between (56) and (58) to get

$$\neg F^-(p) \vee \neg F^-(p-1) \vee G^+(p-1) \vee G^-(p),$$

and then use this together with (53.1), …, (53.$p-2$), (55), (53.$p+1$), …, (53.$q-1$) to work-out the goal in (34.$q$).

**Proof of Lemma 8.3** Let $c$ be the string in $\{0,1\}^B$ such that $b = a + c$. We aim for a proof of

$$\frac{\mathrm{CARRY}_{k,\ell}(\mathbf{x}, a)}{\mathrm{CARRY}_{k,\ell}(\mathbf{x}, a, c)} \tag{59}$$

and then apply Lemma 8.2 on the result to get

$$\mathrm{CARRY}_{k,\ell}(\mathbf{x}, b).$$

Let us prove (59). For the sake of argument, let $M_1$ denote the automaton with input $\mathbf{x}, a$ and let $M_2$ denote the automaton with input $\mathbf{x}, a, c$. Intuitively, what we want to prove that for every $p \in \{1, \ldots, B\}$ either $N_\ell(\mathbf{x}, a, c; p) \geq 0$, which means that $M_2$ accepts at time $p$, or $-k < N_\ell(\mathbf{x}, a, c; p) < 0$ and $S_\ell(\mathbf{x}, a, c; p) \geq S_\ell(\mathbf{x}, a; p)$, which means $M_2$ at time $p$ is still at an intermediate state but not falling behind the state of $M_1$ at time $p$. If we succeed in proving this, then the assumption that $M_1$ eventually accepts will imply that $M_2$ eventually accepts as well. We proceed by induction on $p$. For later reference we state the base cases and inductive cases as claims:

**Claim 8.** $S_\ell(\mathbf{x}, a, c; 0) \geq S_\ell(\mathbf{x}, a; 0)$.

*Proof.* This is immediate from the fact that $S(\mathbf{x}, a, c; 0) = S(\mathbf{x}, a; 0) = -1$ by definition. □

**Claim 9.** *Let* $1 \leq p \leq B$.
*If both*

1. $-k < N_\ell(\mathbf{x}, a; p) < 0$, *and*
2. $S_\ell(\mathbf{x}, a, c; p-1) \geq S_\ell(\mathbf{x}, a; p-1)$,

*then either*

1. $N_\ell(\mathbf{x}, a, c; p) \geq 0$, *or*
2. $-k < N_\ell(\mathbf{x}, a, c; p) < 0 \wedge S_\ell(\mathbf{x}, a, c; p) \geq S_\ell(\mathbf{x}, a; p)$.

*Proof.* If $N_\ell(\mathbf{x}, a, c; p) \geq 0$ there is nothing to prove. Assume then $N_\ell(\mathbf{x}, a, c; p) < 0$. The assumption $-k < N_\ell(\mathbf{x}, a; p) < 0$ together with Lemma 6 gives $S_\ell(\mathbf{x}, a; p) = N_\ell(\mathbf{x}, a; p)$. On the other hand,

$$\begin{aligned}
N_\ell(\mathbf{x}, a, c; p) &= 2S_\ell(\mathbf{x}, a, c; p-1) + (\mathbf{x}_p + a_p + c_p) \\
&\geq 2S_\ell(\mathbf{x}, a; p-1) + (\mathbf{x}_p + a_p) + c_p \\
&= N_\ell(\mathbf{x}, a; p) + c_p \\
&\geq N_\ell(\mathbf{x}, a; p),
\end{aligned}$$

where the first inequality follows from the assumption $S_\ell(\mathbf{x}, a, c; p-1) \geq S_\ell(\mathbf{x}, a; p-1)$, and the second inequality follows from the fact that $c_p \geq 0$. From the assumption $-k < N_\ell(\mathbf{x}, a; p) < 0$ we conclude that $N_\ell(\mathbf{x}, a, c; p) > -k$. At this point we have $-k < N_\ell(\mathbf{x}, a, c; p) < 0$ and we can apply Lemma 6 to obtain $S_\ell(\mathbf{x}, a, c; p) = N_\ell(\mathbf{x}, a, c; p)$. Putting all these together we get $S_\ell(\mathbf{x}, a, c; p) \geq S_\ell(\mathbf{x}, a; p)$. □

**Claim 10.** *Let* $1 \leq p \leq B$.
*If both*

1. $N_\ell(\mathbf{x}, a; p) \geq 0$, *and*
2. $S_\ell(\mathbf{x}, a, c; p-1) \geq S_\ell(\mathbf{x}, a; p-1)$,

*then*

$$N_\ell(\mathbf{x}, a, c; p) \geq 0.$$

*Proof.* We have

$$\begin{aligned} N_\ell(\mathbf{x}, a, c; p) &= 2S_\ell(\mathbf{x}, a, c; p-1) + (\mathbf{x}_p + a_p + c_p) \\ &\geq 2S_\ell(\mathbf{x}, a; p-1) + (\mathbf{x}_p + a_p) + c_p \\ &= N_\ell(\mathbf{x}, a; p) + c_p \\ &\geq N_\ell(\mathbf{x}, a; p), \end{aligned}$$

where the first inequality follows from the assumption $S_\ell(\mathbf{x}, a, c; p-1) \geq S_\ell(\mathbf{x}, a; p-1)$, and the second inequality follows from the fact that $c_p \geq 0$. From the assumption $N_\ell(\mathbf{x}, a; p) \geq 0$ we conclude that $N_\ell(\mathbf{x}, a, c; p) \geq 0$. □

Turning this meta-level argument into a formal proof is a matter of choosing the right notation. For every $p \in \{0, \ldots, B\}$ define a formula:

$$R(p) \quad \equiv \quad S_\ell(\mathbf{x}, a, c; p) \geq S_\ell(\mathbf{x}, a; p).$$

For every $p \in \{1, \ldots, B\}$ define formulas:

$$\begin{array}{rcl} F^+(p) & \equiv & N_\ell(\mathbf{x}, a; p) \geq 0 \\ G^+(p) & \equiv & N_\ell(\mathbf{x}, a, c; p) \geq 0 \\ F^-(p) & \equiv & -k < N_\ell(\mathbf{x}, a; p) < 0 \\ G^-(p) & \equiv & -k < N_\ell(\mathbf{x}, a, c; p) < 0. \end{array}$$

Note that the formulas $\text{CARRY}_{k,\ell}(\mathbf{x}, a)$ and $\text{CARRY}_{k,\ell}(\mathbf{x}, a, c)$ are precisely

$$(\exists p : 1 \leq p \leq B)(F^+(p) \wedge (\forall q : 1 \leq q \leq p-1)(F^-(q))) \tag{60}$$

$$(\exists p : 1 \leq p \leq B)(H^+(p) \wedge (\forall q : 1 \leq q \leq p-1)(H^-(q))) \tag{61}$$

Claims 8, 9 and 10 state that all the base-case, inductive-case and terminating-case formulas in the scheme of induction Lemma 10 below are tautologies. Since these are constant-size $\Sigma_{1,\mathsf{c}}$-formulas, by completeness they have $\Sigma_{1,\mathsf{c}}$-Frege proofs of constant size. The proof of (59) follows from this scheme applied to these formulas and to the hypothesis formula (60).

**Lemma 10.** *The following assertion has polynomial-size $\Sigma_{2,\mathsf{c}}$-Frege proofs:*
*Given the* base-case*:*

$R(0)$,

*and, for every $p \in \{1, \ldots, B\}$, the* inductive-case *and* terminating-case*:*

$\neg F^-(p) \vee \neg R(p-1) \vee H^+(p) \vee H^-(p)$,

$\neg F^-(p) \vee \neg R(p-1) \vee H^+(p) \vee R(p)$,

$\neg F^+(p) \vee \neg R(p-1) \vee H^+(p)$,

*and given the* hypothesis*:*

$(\exists p : 1 \leq p \leq B)(F^+(p) \wedge (\forall q : 1 \leq q \leq p-1)(F^-(q)))$,

*conclude:*

$(\exists p : 1 \leq p \leq B)(H^+(p) \wedge (\forall q : 1 \leq q \leq p-1)(H^-(q)))$.

*Proof.* This is a very special case of the previous scheme of induction where $S(p) = R(p) = T(p)$ for every $p \in \{0, 1, \ldots, B\}$, and $G^-(p) = F^-(p)$ and $G^+(p) = F^+(p)$ for every $p \in \{1, \ldots, B\}$. □

**Proof of Lemma 8.4** Intuitively we want to prove that $N_\ell(z, \overline{z}; p) \leq -1$ for every $p \in \{1, \ldots, B\}$. This will be in contradiction with the hypothesis $\text{CARRY}_{k,\ell}(z, \overline{z})$, which states that there exists some $p \in \{1, \ldots, B\}$ such that $N_\ell(z, \overline{z}; p) \geq 0$, and this is what we want. The proof that $N_\ell(z, \overline{z}; p) \leq -1$ is essentially direct from the definitions:

**Claim 11.** *Let $p \in \{1, \ldots, B\}$. Then $N_\ell(z, \overline{z}; p) \leq -1$.*

*Proof.* Since $S_\ell(z, \overline{z}; p-1) = (A_\ell(z, \overline{z}; p-1) \mod 2^{p'}) - 2^{p'}$ for $p' = \min\{p-1, \ell\}$, automatically $S_\ell(z, \overline{z}; p-1) \leq -1$. Therefore

$$N_\ell(z, \overline{z}; p) = 2S_\ell(z, \overline{z}; p-1) + z_p + 1 - z_p \leq -2 + 1 = -1.$$

□

Using the notation $F^+(p) = F^+(z, \overline{z}; p)$, this claim shows that $\neg F^+(p)$ is a tautology for every $p \in \{1, \ldots, B\}$. Since this is a constant-size $\Sigma_{1,\mathsf{c}}$-formula, by completeness it must have a $\Sigma_{1,\mathsf{c}}$-Frege proof of constant size. Weakening on it gives

$$\neg F^+(p) \vee (\exists q : 1 \leq q \leq p-1)(\neg F^-(q)). \tag{62}$$

The proof of Lemma 84 is completed by a sequence of CUTs, starting with a CUT between the hypothesis $\text{CARRY}_{k,\ell}(z, \overline{z})$ and (62.1), then a CUT between the result and (62.2), and so on until we use (62.$B$). At that point we will have derived the empty clause □.

# 6 Simulating chaining refutations

In this section we use the CARRY formula with parameters $k = 11$, $\ell = 4$ and $B = M+2$, where $M$ is a large integer, that we think of as unbounded. As $k$ and $\ell$ stay fixed everywhere in the section, for convenience we write CARRY instead of $\text{CARRY}_{11,4}$. Note that CARRY is a $\Sigma_{2,44}$-formula.

The letters $x$, $y$ and $z$ denote integer variables ranging over $[0, 2^M)$, and $X$, $Y$ and $Z$ denote strings of $M$ Boolean variables for the binary representations of $x$, $y$ and $z$. The letters $a$, $b$ and $c$ denote integer constants in the range $(-2^M, 2^M)$, and $A$, $B$ and $C$ denote bit-strings of length $M$ for the binary representations of their absolute values $|a|$, $|b|$ and $|c|$. For an integer $d$ in $[0, 2^M)$, we use the notation $\overline{d} = \overline{d}_M$ to the denote the integer $2^M - 1 - d$. Note that $\overline{d}$ is also an integer in $[0, 2^M)$. Moreover the binary representation of $\overline{d}$ with $M$ bits is precisely the bit-wise complement of the binary representation of $d$ with $M$ bits. This justifies the notation $\overline{d}$.

## 6.1 Representing atoms and max-atoms

An atom is an expression of the form $x \leq y + a$. We distinguish *positive atoms* of the type $x \leq y + a$ with $a \geq 0$ from *negative atoms* of the type $x \leq y - a$ with $a \geq 0$.

**Positive atoms** First note that $x \leq y + a$ is equivalent to $2^M \leq \overline{x} + y + a + 1$ since $x + \overline{x} = 2^M - 1$. For later use we add $3 \cdot 2^M$ to both sides of this inequality to get:

$$2^{M+2} \leq \overline{x} + y + a + 3 \cdot 2^M + 1.$$

Interpreting bit-strings as the non-negative integers in binary this is represented by

$$\text{CARRY}(00\overline{X}, 00Y, 00A, 010^M, 010^M, 010^M, 0^M01).$$

Note how we padded the strings so that each has length $M + 2$.

**Negative atoms** Note that $x \leq y - a$ is equivalent to $2^{M+1} \leq \overline{x} + y + \overline{a} + 1 + 1$ since $x + \overline{x} = 2^M - 1$ and $a + \overline{a} = 2^M - 1$. For later use we add $2 \cdot 2^M$ to both sides of this inequality to get:

$$2^{M+2} \leq \overline{x} + y + \overline{a} + 2 \cdot 2^M + 1 + 1.$$

Interpreting bit-strings as non-negative integers in binary this is represented by

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00\overline{A}, 010^M, 010^M, 0^M01, 0^M01).$$

Note how we padded the strings so that each has length $M + 2$.

**Intermediate atoms** For technical reason in the proofs we need to view expressions of the form $x \leq y + a + b$ as different from $x \leq y + c$ where $c = a + b$. We distinguish the four cases:

1. $x \leq y + a + b$,
2. $x \leq y + a - b$,
3. $x \leq y - a + b$,
4. $x \leq y - a - b$,

where in such cases both $a$ and $b$ are non-negative integers. By the same reasoning as before the four expressions are represented by:

1. $\mathrm{CARRY}(00\overline{X}, 00Y, 00A, 00B, 010^M, 010^M, 010^M, 0^M01)$,
2. $\mathrm{CARRY}(00\overline{X}, 00Y, 00A, 00\overline{B}, 010^M, 010^M, 0^M01, 0^M01)$,
3. $\mathrm{CARRY}(00\overline{X}, 00Y, 00\overline{A}, 00B, 010^M, 010^M, 0^M01, 0^M01)$,
4. $\mathrm{CARRY}(00\overline{X}, 00Y, 00\overline{A}, 00\overline{B}, 010^M, 0^M01, 0^M01, 0^M01)$.

**Max-atoms** Let $I$ be the max-atom $x \leq \max\{x_1 + a_1, \ldots, x_r + a_r\}$, where all constants are in the range $(-2^M, 2^M)$. We represent $I$ by the formula

$$(\exists i : 1 \leq i \leq r)(x \leq x_i + a_i).$$

Note that this is again a $\Sigma_{2,44}$-formula. We write $F(I) = F_M(I)$ for this formula, for the indicated value of the parameter $M$. If $\mathcal{I}$ is a collection of max-atoms, we write $F(\mathcal{I}) = F_M(\mathcal{I})$ for the collection of all $F(I)$ as $I$ ranges over $\mathcal{I}$.

## 6.2 Inferences with atoms

We start giving proofs of some basic assertions for atoms. These will serve as base for max-atoms.

**Lemma 11.** *The following assertions have polynomial-size $\Sigma_{2,\mathsf{c}}$-Frege proofs:*

1. *given $x \leq z + a$ and $z \leq y + b$ conclude $x \leq y + a + b$,*
2. *given $x \leq y + a + b$ conclude $x \leq y + c$, if $c = a + b$,*
3. *given $x \leq y + a$ conclude $x \leq y + b$, if $a \leq b$.*

**Proof of Lemma 11.1** We need to distinguish three cases according to the signs of $a$ and $b$ (and by the symmetry between $a$ and $b$). Let $A$ and $B$ be the bit-strings of length $M$ that represent $|a|$ and $|b|$ in binary notation.

Consider first the case $a \geq 0$, $b < 0$. The two hypothesis $x \leq z+a$ and $z \leq y+b$ are represented by:

$$\mathrm{CARRY}(00\overline{X}, 00Z, 00A, 010^M, 010^M, 010^M, 0^M01),$$
$$\mathrm{CARRY}(00\overline{Z}, 00Y, 00\overline{B}, 010^M, 010^M, 0^M01, 0^M01).$$

Applying first Lemma 8.2 to the first, then Lemma 7.5 to both, and simplifying with Lemma 7.1, we get:

$$\mathrm{CARRY}(00\overline{X}, 10Z, 00A, 010^M, 0^M01),$$
$$\mathrm{CARRY}(01\overline{Z}, 00Y, 00\overline{B}, 010^M, 0^M01, 0^M01).$$

Now note that $10Z$ and $01\overline{Z}$ are complementary strings so we can apply Lemma 8.1 to get

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00A, 00\overline{B}, 010^M, 010^M, 0^M01, 0^M01).$$

This is eactly the representation of $x \leq y+a+b$, for the case $a \geq 0, b < 0$.

Next, consider the case $a, b \geq 0$. The two hypothesis $x \leq z+a$ and $z \leq y+b$ are represented by:

$$\mathrm{CARRY}(00\overline{X}, 00Z, 00A, 010^M, 010^M, 010^M, 0^M01),$$
$$\mathrm{CARRY}(00\overline{Z}, 00Y, 00B, 010^M, 010^M, 010^M, 0^M01).$$

Applying to the first hypothesis Lemma 8.2 twice, then Lemma 7.5, and simplifying with Lemma 7.1, we get:

$$\mathrm{CARRY}(00\overline{X}, 11Z, 00A, 0^M01),$$
$$\mathrm{CARRY}(00\overline{Z}, 00Y, 00B, 010^M, 010^M, 010^M, 0^M01).$$

Since $11Z$ and $00\overline{Z}$ are complementary strings we can apply Lemma 8.1 to get

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00A, 00B, 010^M, 010^M, 010^M, 0^M01).$$

This is eactly the representation of $x \leq y+a+b$, for the case $a, b \geq 0$.

Finally, consider the case $a, b < 0$. The two hypothesis $x \leq z+a$ and $z \leq y+b$ are represented by:

$$\mathrm{CARRY}(00\overline{X}, 00Z, 00\overline{A}, 010^M, 010^M, 0^M01, 0^M01),$$
$$\mathrm{CARRY}(00\overline{Z}, 00Y, 00\overline{B}, 010^M, 010^M, 0^M01, 0^M01).$$

Applying to the first hypothesis Lemma 8.2, then applying to both Lemma 7.5 and simplifying with Lemma 7.1, we get:

$$\mathrm{CARRY}(00\overline{X}, 10Z, 00\overline{A}, 0^M01, 0^M01),$$
$$\mathrm{CARRY}(01\overline{Z}, 00Y, 00\overline{B}, 010^M, 0^M01, 0^M01).$$

Since $10Z$ and $01\overline{Z}$ are complementary strings we can apply Lemma 8.1 to get

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00\overline{A}, 00\overline{B}, 010^M, 0^M01, 0^M01, 0^M01).$$

This is eactly the representation of $x \leq y+a+b$, for the case $a, b < 0$.

**Proof of Lemma 11.2** We need to distinguish four cases according to the signs of $a$, $b$, and $c$. Let $A$, $B$, and $C$ be the bit-strings of length $M$ that represent $|a|$, $|b|$, and $|c|$ in binary, respectively.

Consider first the case $a < 0$, $b \geq 0$ and $c < 0$, which implies $C = A - B$. The hypothesis $x \leq y + a + b$ is represented by

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00\overline{A}, 00B, 010^M, 010^M, 0^M01, 0^M01).$$

Applying Lemma 8.2 to the numbers $00\overline{A}$ and $00B$, since we have $00\overline{A} + 00B = 001^M - 00A + 00B = 001^M - 00C = 00\overline{C}$, we get

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00\overline{C}, 010^M, 010^M, 0^M01, 0^M01).$$

This is exactly the representation of $x \leq y + c$, when $c < 0$.

Second, we consider the case $a < 0$, $b \geq 0$ and $c \geq 0$, which implies $C = B - A$. The hypothesis is the same as in the previous case. Applying Lemma 8.2 first to the numbers $00\overline{A}$ and $00B$, then to the resulting string together with $0^M01$, and noting that $00\overline{A} + 00B + 0^M01 = (001^M - 00A) + 00B + 0^M01 = 010^M + 00C = 01C$, we get

$$\mathrm{CARRY}(00\overline{X}, 00Y, 01C, 010^M, 010^M, 0^M01).$$

By applying Lemma 7.2 and Lemma 7.5 to the above we get

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00C, 010^M, 010^M, 010^M, 0^M01).$$

This is exactly the representation of $x \leq y + c$, when $c \geq 0$.

Third, we consider the case $a, b, c < 0$, which implies $C = A + B$. The hypothesis $x \leq y + a + b$ is represented by

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00\overline{A}, 00\overline{B}, 010^M, 0^M01, 0^M01, 0^M01).$$

Applying Lemma 8.2 first to the numbers $00\overline{A}$ and $00\overline{B}$, then to the resulting string together with $0^M01$, and noting that $00\overline{A} + 00\overline{B} + 0^M01 = (001^M - 00A) + (001^M - 00B) + 0^M01 = 010^M + 00\overline{C} = 01\overline{C}$, we get

$$\mathrm{CARRY}(00\overline{X}, 00Y, 01\overline{C}, 010^M, 0^M01, 0^M01).$$

By applying Lemma 7.2 and Lemma 7.5 we get

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00\overline{C}, 010^M, 010^M, 0^M01, 0^M01).$$

This is exactly the representation of $x \leq y + c$, when $c < 0$.

Finally, we consider the case $a, b, c \geq 0$, which implies $C = A + B$. The hypothesis $x \leq y + a + b$ is represented by

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00A, 00B, 010^M, 010^M, 010^M, 0^M01).$$

Applying Lemma 8.2 to $00A$ and $00B$, we get

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00C, 010^M, 010^M, 010^M, 0^M01).$$

This is exactly the representation of $x \leq y + c$, when $c \geq 0$.

**Proof of Lemma 11.3** We may assume $a < b$. We need to distinguish three cases according to the signs of $a$ and $b$. Let $A$ and $B$ be the bit-strings of length $M$ that represent $|a|$ and $|b|$ respectively.

Consider first the case $a, b \geq 0$ which implies $A < B$. The hypothesis $x \leq y + a$ is represented by

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00A, 010^M, 010^M, 010^M, 0^M01).$$

Applying Lemma 8.3 for $00A < 00B$ we get

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00B, 010^M, 010^M, 010^M, 0^M01).$$

This is exactly the representation of $x \leq y + b$, when $b \geq 0$.

Next, consider the case $a < 0$, $b \leq 0$ which implies $B < A$. The hypothesis $x \leq y + a$ is represented by

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00\overline{A}, 010^M, 010^M, 0^M01, 0^M01).$$

Applying Lemma 8.3 for $00\overline{A} < 00\overline{B}$ we get

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00\overline{B}, 010^M, 010^M, 0^M01, 0^M01).$$

This is exactly the representation of $x \leq y + b$, when $b < 0$. In the special case of $b = 0$, we apply Lemma 7.2 to introduce the term $00B = 000^M$ and then Lemma 8.2 to the strings $00\overline{B} = 001^M$ and $0^M01$ to get:

$$\mathrm{CARRY}(00\overline{X}, 00Y, 00B, 010^M, 010^M, 010^M, 0^M01).$$

which is the representation of $x \leq y + b$ for $b = 0$.

Finally, the third case $a < 0$, $b > 0$ follows from the two assertions: given $x \leq y + a$ conclude $x \leq y + 0$, and given $x \leq y + 0$ conclude $x \leq y + b$.

## 6.3 Inferences with max-atoms

We are ready to simulate the rules of the chaining inference system. In the following, the letters $R$, $S$ and $T$ denote sets of offsets of the form $x + c$ for a variable $x$ and a constant $c$. The notation $S + a$ refers to the collection of offsets of the form $x + (c + a)$ as $x + c$ ranges over all offsets in $S$. The three assertions in the following lemma correspond to the three rules of chaining:

**Lemma 12.** *The following assertions have polynomial-size $\Sigma_{2,\mathsf{c}}$-Frege proofs:*

1. *given $x \leq \max(R, y + a)$ and $y \leq \max(S)$ conclude $x \leq \max(R, T)$, if $T = S + a$,*
2. *given $x \leq \max(R, x + a)$ conclude $x \leq \max(R)$, if $a < 0$,*
3. *given $x \leq \max(R, y + a, y + b)$ conclude $x \leq \max(R, y + c)$, if $a \leq c$ and $b \leq c$.*

For the coming proofs, let $R$ be $\{z_i + a_i : i \in I\}$ and $S$ be $\{z_j + b_j : j \in J\}$.

**Proof of Lemma 12.1** Recall that $x \leq \max(R, y+a)$ and $y \leq \max(S)$ are abbreviations for disjunctions of atoms. If we show that for every atom $A$ in $x \leq \max(R, y+a)$ and every atom $B$ in $y \leq \max(S)$ we have a proof of

$$\frac{A \quad B}{x \leq \max(R, S+a)}, \tag{63}$$

the rest will follow from the proof-scheme of pairwise case analysis Lemma 3. If the atom $A$ is an atom of the form $x \leq z_i + a_i$ for some $i \in I$, then $A$ is also in the conclusion of (63). In this case the proof is a single application of weakening on $A$. Let us assume then that $A$ is the atom $x \leq y + a$. Let $B$ the the atom $y \leq z_j + b_j$ for some $j \in J$. Lemma 11.1 on $A$ and $B$ gives $x \leq z_j + b_j + a$, and Lemma 11.2 on it gives the atom $x \leq z_j + (b_j + a)$. This atom is in the conclusion of (63) and an application of weakening on it gives the proof of (63).

**Proof of Lemma 12.2** Recall that $x \leq \max(R, x+a)$ is an abbreviation for a disjunction of atoms. If we show that for every atom $A$ in $x \leq \max(R, x+a)$ we have a proof of

$$\frac{A}{x \leq \max(R)}, \tag{64}$$

the rest will follow from the proof-scheme of case analysis Lemma 4. If the atom $A$ is an atom of the form $x \leq z_i + a_i$ for some $i \in I$, then $A$ is also in the conclusion of (64). In this case the proof is a single application of weakening on $A$. Let us assume then that $A$ is the atom $x \leq x + a$. If $a \leq -1$, Lemma 11.3 on $A$ gives $x \leq x - 1$. This atom is represented by

$$\mathrm{CARRY}(00\overline{X}, 00X, 001^{M-1}0, 010^M, 010^M, 0^M01, 0^M01).$$

Note that $001^{M-1}0 + 010^M + 010^M + 0^M01 + 0^M01 = 110^M$. Therefore, four applications of Lemma 8.2 give

$$\mathrm{CARRY}(00\overline{X}, 00X, 110^M).$$

Two applications of Lemma 7.5, one application of Lemma 7.2, and one application of Lemma 8.4 give $\Box$. Weakening on it gives the conclusion in (64).

**Proof of Lemma 12.3** Recall that $x \leq \max(R, y+a, y+b)$ is an abbreviation for a disjunction of atoms. If we show that for every atom $A$ in $x \leq \max(R, y+a, y+b)$ we have a small proof of

$$\frac{A}{x \leq \max(R, y+c)}, \tag{65}$$

the rest will follow from the proof-scheme of case analysis Lemma 4. If the atom $A$ is an atom of the form $x \leq z_i + a_i$ for some $i \in I$, then $A$ is also in the conclusion of (64). In this case the proof is a single application of weakening on $A$. Let us assume then that $A$ is the atom $x \leq y + a$; the case $x \leq y + b$ is analogous. If $a \leq c$, Lemma 11.3 gives $x \leq y + c$, which is actually an atom in the conclusion of (65). Weakening on $x \leq y + c$ gives the proof.

## 7 Main result and consequences

In this section we state the main result and its consequences for propositional proof-complexity.

### 7.1 Main result

Before we state it we need to recall two standard tricks in proof-complexity.

**Converting to 3-CNF formulas** For a Boolean formula $F = F(x)$ with variables $x = x_1 \cdots x_n$, let $T = T(x, y) = T_3(F)$ denote the standard translation of $F$ into a 3-CNF-formula with the same variables $x$ and possibly additional variables $y = y_1 \cdots y_m$. This formula has the property that for every $a \in \{0,1\}^n$ the following equivalence holds:

$$F(a) = 1 \text{ if and only if } \text{ there exists } b \in \{0,1\}^m \text{ such that } T(a,b) = 1.$$

If the size of $F$ is at most $s$, then the number of additional variables $y$ is at most $2s$, and the number of clauses in $T$ is at most $4s$. Also, if $F$ is a $\Sigma_{d,k}$-formula, then the assertion $T \vdash F$ has a polynomial-size $\Sigma_{d,k}$-Frege proof.

**Effectively simulating bottom fan-in** The second trick concerns the relationship between $\Sigma_{d,k}$-Frege and $\Sigma_d$-Frege. This trick was used in [2] for $d = 1$ and is called *effective simulation* in [23].

In general, it is not true that $\Sigma_d$-Frege polynomially simulates $\Sigma_{d,k}$-Frege. For example, it is known that $\Sigma_1$-Frege does not polynomially simulate $\Sigma_{1,2}$-Frege [3]. However, it *effectively simulates* it. The idea is that if $\mathcal{C}$ is a set of clauses on the variables $x_1, \ldots, x_n$, we can add additional variables $z_T$ and $z_C$ for every possible term $T$ and clause $C$ of at most $k$ literals on the variables $x_1, \ldots, x_n$, and axioms that fix the truth value of the new variables accordingly:

$$\begin{array}{llll} (1) & z_{C_1 \vee C_2} \leftrightarrow z_{C_1} \vee z_{C_2} & (3) & z_{x_i} \leftrightarrow x_i \\ (2) & z_{T_1 \wedge T_2} \leftrightarrow z_{T_1} \wedge z_{T_2} & (4) & z_{\overline{x_i}} \leftrightarrow \overline{x_i} \end{array}$$

Let $E_k(\mathcal{C})$ be the extension of $\mathcal{C}$ with these axioms converted to clauses. Note that if $\mathcal{C}$ is satisfiable, then $E_k(\mathcal{C})$ stays satisfiable: set $z_C$ and $z_T$ to the truth-value of $C$ and $T$ under the truth-assignment satisfying $\mathcal{C}$. On the other hand, if $\mathcal{C}$ is unsatisfiable, the size of the smallest refutation of $\mathcal{C}$ in $\Sigma_{d,k}$-Frege is polynomially related to the size of the smallest refutation of $E_k(\mathcal{C})$ in $\Sigma_d$-Frege. Moreover, there are efficient conversions from one to the other. In particular, all this implies that the weak automatizability problem for $\Sigma_{d,k}$-Frege reduces to the one for $\Sigma_d$-Frege.

With this notation we can state the main result of the paper. In the statement of this result, the unspecified universal constant in $E_{\mathsf{c}}$ is the one from Lemma 12.

**Theorem 1.** *Let $\mathcal{G}$ be a mean-payoff game with $v$ vertices and weights in the range $[-W, W]$. Let $M$ be an integer such that $2^M > Wv^2$. Let $\mathcal{C} = E_{\mathsf{c}}(T_3(F_M(I(\mathcal{G}))))$. The following hold:*

1. *if $\nu_{\mathcal{G}} \geq 0$, then $\mathcal{C}$ is satisfiable,*
2. *if $\nu_{\mathcal{G}} < 0$, then $\mathcal{C}$ has polynomial-size $\Sigma_2$-Frege refutations.*

*Proof.* This is a direct consequence of Lemma 1, Lemma 12, and the tricks above. □

It is perhaps worth noting that the $\Sigma_2$-refutation in this theorem is actually a Res($B$)-refutation, where $B = M + 2$ and $M = \lceil 2\log_2(v) + \log_2(W) \rceil$. The reason is that each max-atom is a disjunction of CARRY-formulas with parameter $B$, and each CARRY-formula with parameter $B$ is a disjunction of conjunctions of fan-in $B$, with constant fan-in disjunctions at the bottom that end-up wiped away by the $E_{\mathsf{c}}$-trick. Since the size of $\mathcal{C}$ is polynomial in $v^2 \log_2(W)$, this is slightly better than a plain polynomial-size $\Sigma_2$-refutation as stated in the theorem.

### 7.2 Consequences for automatizability and interpolation

In this section, let $\mathcal{G}$, $v$, $W$, $M$ and $\mathcal{C}$ be as in the statement of Theorem 1.

One immediate consequence of Theorem 1 is that if $\Sigma_2$-Frege were weakly automatizable, there would be a polynomial-time algorithm for solving mean-payoff games. Indeed, the statement itself of Theorem 1 is a reduction from MPG to the weak automatizability problem for $\Sigma_2$-Frege. Clearly this reduction is computable in polynomial time.

On the other hand, there is a tight connection between weak automatizability, interpolation, and the provability of the *reflection principle* (see [27]). We discuss this briefly. Let $\mathrm{SAT}_{n,m}(x, y)$ be a CNF-formula that expresses that $y$ is an assignment satisfying the CNF-formula encoded by $x$. Here $n$ and $m$ are the number of variables and the number of clauses of the formula encoded by $x$. Let $\mathrm{REF}_{n,m,r,d}(x, z)$ be a CNF formula that expresses that $z$ is the encoding of a $\Sigma_d$-refutation of the CNF-formula encoded by $x$. Here $n$ and $m$ are as in $\mathrm{SAT}_{n,m}$, and $r$ is the size of the proof encoded by $z$. Formalizing this requires some standard encoding of CNF-formulas, $\Sigma_d$-formulas, and $\Sigma_d$-Frege proofs. Obviously, the formula

$$\mathrm{SAT}_{n,m}(x, y) \wedge \mathrm{REF}_{n,m,r,d}(x, z) \tag{66}$$

is unsatisfiable. This is called the reflection principle for $\Sigma_d$-Frege. It turns out that (66) has a polynomial-size refutation in $\Sigma_{d,2}$-Frege. This was observed in [2] for $d = 1$ and the proof can be extended to bigger $d$ in a natural way.

It follows that if $\Sigma_{2,2}$-Frege enjoyed feasible interpolation, there would be an algorithm for solving mean-payoff games in polynomial time. Indeed, the reduction from MPG to the interpolation problem for $\Sigma_{2,2}$ goes as follows: given a game $\mathcal{G}$, it suffices to run the interpolation algorithm fed with a refutation of (66) and the setting of $x$ to the encoding of the CNF-formula $\mathcal{C}$. Of course we choose $n$ and $m$ to be the number of variables and clauses of $\mathcal{C}$, and $r$ and $d$ to be the size of the $\Sigma_2$-Frege proof of $\mathcal{C}$ and 2. By Theorem 1 exactly one of $\mathrm{SAT}(\mathcal{C}, y)$ or $\mathrm{REF}(\mathcal{C}, z)$ is satisfiable, which means that the interpolation algorithm will return the other. This will tell us whether $\nu_{\mathcal{G}} \geq 0$ or $\nu_{\mathcal{G}} < 0$.

We state these two observations as a corollary:

**Corollary 1.** *There exists a polynomial-time reduction from MPG to the weak automatizability problem for $\Sigma_2$-Frege, and to the interpolation problem for $\Sigma_{2,2}$-Frege.*

An intriguing question is whether a reverse connection exists. Clearly the weak automatizability problem for a proof system is related to proving proof-size lower bounds for it, and the latter has an obvious game-theoretic flavour. In this context it is perhaps interesting to recall that Zwick and Paterson modeled the complexity of selection and sorting algorithms with limited storage as mean-payoff games between an algorithm designer and an adversary [29]. Perhaps the proof-search problem for a natural proof system for propositional logic can also be cast in such terms.

**Acknowledgements.** We thank Manuel Bodirsky for bringing [21] to our attention. This work was done while visiting the Centre de Recerca Matemàtica (CRM), Bellaterra, Barcelona.

## References

[1] M. Alekhnovich and A. A. Razborov. Resolution is not automatizable unless W[P] is tractable. In *42nd Annual IEEE Symposium on Foundations of Computer Science*, pages 210–219, 2001.

[2] A. Atserias and M. L. Bonet. On the automatizability of resolution and related propositional proof systems. *Information and Computation*, 189(2):182–201, 2004.

[3] A. Atserias, M. L. Bonet, and J. L. Esteban. Lower bounds for the weak pigeonhole principle and random formulas beyond resolution. *Information and Computation*, 176(2):136–152, 2002. A preliminary version appeared in ICALP 2001.

[4] A. Atserias and E. Maneva. Mean payoff-games and the max-atom problem. Technical report, http://www.lsi.upc.edu/˜atserias, 2009.

[5] P. Beame and T. Pitassi. Simplified and improved resolution lower bounds. In *37th Annual IEEE Symposium on Foundations of Computer Science*, pages 274–282, 1996.

[6] P. Beame and T. Pitassi. Propositional proof complexity: Past, present, and future. In *The Bulletin of EATCS*, volume 65, pages 66–89. EATCS, 1998.

[7] E. Ben-Sasson and A. Wigderson. Short proofs are narrow–resolution made simple. *Journal of the ACM*, 48(2):149–169, 2001. A preliminary version appeared in STOC'99.

[8] M. Bezem, R. Nieuwenhuis, and E. Rodríguez-Carbonell. The max-atom problem and its relevance. In *Proceedings of 15th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning*, volume 5330 of *Lecture Notes in Computer Science*, pages 47–61. Springer, 2008.

[9] H. Björklund and S. Vorobyov. Combinatorial structure and randomized subexponential algorithms for infinite games. *Theoretical Computer Science*, 349(3):347–360, 2005.

[10] M. L. Bonet, C. Domingo, R. Gavaldà, A. Maciel, and T. Pitassi. Non-automatizability of bounded-depth frege proofs. *Computational Complexity*, 13:47–68, 2004. A preliminary version appeared in CCC'99.

[11] M. L. Bonet, T. Pitassi, and R. Raz. Lower bounds for cutting planes proofs with small coefficients. *Journal of Symbolic Logic*, 62(3):708–728, 1997. A preliminary version appeared in STOC'95.

[12] M. L. Bonet, T. Pitassi, and R. Raz. On interpolation and automatization for Frege systems. *SIAM Journal of Computing*, 29(6):1939–1967, 2000. A preliminary version appeared in FOCS'97.

[13] A. Ehrenfeucht and J. Mycielsky. Positional strategies for mean payoff games. *International Journal of Game Theory*, 8(2):109–113, 1979.

[14] M. Furst, J. Saxe, and M. Sipser. Parity, circuits and the polynomial time hierarchy. *Mathematical Systems Theory*, 17:13–27, 1984.

[15] M. Jurdziński. Deciding the winner in parity games is in UP ∩ co-UP. *Information Processing Letters*, 68:119–124, 1998.

[16] M. Jurdziński, M. Paterson, and U. Zwick. A deterministic subexponential algorithm for solving parity games. *SIAM Journal on Computing*, 38(4):1519–1532, 2008. A preliminary version appeared in SODA'06.

[17] R. M. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Mathematics*, 23(3):309–311, 1978.

[18] J. Krajícek. Interpolation theorems, lower bounds for proof systems, and independence results for bounded arithmetic. *Journal of Symbolic Logic*, 62:457–486, 1997.

[19] J. Krajícek and P. Pudlák. Some consequences of cryptographical conjectures for $S_2^1$ and $EF$. *Information and Computation*, 140(1):82–94, 1998.

[20] L. Lovász and M. D. Plummer. *Matching Theory*, volume 367. AMS Chelsea Publishing, 2009.

[21] R. H. Möhring, M. Skutella, and F. Stork. Scheduling with AND/OR Precedence Constraints. *SIAM Journal on Computing*, 33(2):393–415, 2004.

[22] C. H. Papadimitriou. *Computational Complexity.* Addison-Wesley, 1995.

[23] T. Pitassi and R. Santhanam. Effectively polynomial simulations. In *Proceedings of First Symposium on Innovations in Computer Science (ICS)*, 2010.

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

[25] P. Pudlák. The lengths of proofs. In *Handbook of proof theory*, pages 547–637. Elsevier Science, 1998.

[26] P. Pudlák. On the complexity of the propositional calculus. In *Sets and Proofs, Invited Papers from Logic Colloquium '97*, pages 197–218. Cambridge University Press, 1999.

[27] P. Pudlák. On reducibility and symmetry of disjoint NP-pairs. *Theoretical Computer Science*, 295:323–339, 2003.

[28] P. Pudlák and J. Sgall. Algebraic models of computation and interpolation for algebraic proof systems. In P. W. Beame and S. R. Buss, editors, *Proof Complexity and Feasible Arithmetic*, volume 39 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 279–296. American Mathematical Society, 1998.

[29] U. Zwick and M. Paterson. The complexity of mean payoff games on graphs. *Theoretical Computer Science*, 158:343–359, 1996.