On the logical complexity of cyclic arithmetic

Das, Anupam

DOI:
10.23638/LMCS-16(1:1)2020

Document Version
Publisher's PDF, also known as Version of record

Citation for published version (Harvard):
Das, A 2020, 'On the logical complexity of cyclic arithmetic', *Logical Methods in Computer Science*, vol. 16, no. 1, 4818, pp. 1:1 - 1:39. https://doi.org/10.23638/LMCS-16(1:1)2020

Logical Methods in Computer Science
Volume 16, Issue 1, 2020, pp. 1:1–1:39
https://lmcs.episciences.org/

Submitted Sep. 12, 2018
Published Jan. 06, 2020

ON THE LOGICAL COMPLEXITY OF CYCLIC ARITHMETIC

ANUPAM DAS

University of Birmingham, United Kindgdom
e-mail address: a.das@bham.ac.uk

Abstract. We study the logical complexity of proofs in cyclic arithmetic (CA), as introduced by Simpson in [Sim17], in terms of quantifier alternations of formulae occurring. Writing $\mathrm{C}\Sigma_n$ for (the logical consequences of) cyclic proofs containing only Σ_n formulae, our main result is that $\mathrm{I}\Sigma_{n+1}$ and $\mathrm{C}\Sigma_n$ prove the same Π_{n+1} theorems, for $n \geq 0$. Furthermore, due to the 'uniformity' of our method, we also show that CA and Peano Arithmetic (PA) proofs of the same theorem differ only exponentially in size.

The inclusion $\mathrm{I}\Sigma_{n+1} \subseteq \mathrm{C}\Sigma_n$ is obtained by proof theoretic techniques, relying on normal forms and structural manipulations of PA proofs. It improves upon the natural result that $\mathrm{I}\Sigma_n \subseteq \mathrm{C}\Sigma_n$. The converse inclusion, $\mathrm{C}\Sigma_n \subseteq \mathrm{I}\Sigma_{n+1}$, is obtained by calibrating the approach of [Sim17] with recent results on the reverse mathematics of Büchi's theorem [KMPS19], and carefully specialising to the case of cyclic proofs. These results improve upon the bounds on proof complexity and logical complexity implicit in [Sim17] and [BT17b].

The uniformity of our method also allows us to recover a metamathematical account of fragments of CA; in particular we show that, for $n \geq 0$, the consistency of $\mathrm{C}\Sigma_n$ is provable in $\mathrm{I}\Sigma_{n+2}$ but not $\mathrm{I}\Sigma_{n+1}$. As a result, we show that certain versions of McNaughton's theorem (the determinisation of ω-word automata) are not provable in RCA_0, partially resolving an open problem from [KMPS19].

1. Introduction

Cyclic and *non-wellfounded* proofs have been studied by a number of authors as an alternative to proofs by induction. This includes cyclic systems for fragments of the modal μ-calculus, e.g. [NW96, SD03, DHL06, DBHS16, Dou17, AL17], structural proof theory for logics with fixed-points, e.g. [San02, FS13, For14, BDS16], (automated) proofs of program termination in separation logic, e.g. [BBC08, BDP11, RB17] and, in particular, cyclic systems for first-order logic with inductive definitions, e.g. [Bro05, Bro06, BS07, BS11]. Due to the somewhat implicit nature of invariants they define, cyclic systems can be advantageous for metalogical analysis, for instance offering better algorithms for proof search, e.g. [BGP12, DP17].

Cyclic proofs may be seen as more intuitively analogous to proofs by 'infinite descent' than proofs by induction (see, e.g., [Sim17]); this subtle difference is enough to make inductive invariants rather hard to generate from cyclic proofs. Indeed it was recently shown that

Key words and phrases: Cyclic proofs, Proof theory, Logical complexity, Peano arithmetic, Induction.
The author is supported by a Marie Skłodowska-Curie fellowship, ERC project 753431.

DOI:10.23638/LMCS-16(1:1)2020

simulating cyclic proofs using induction is not possible for some sub-arithmetic languages [BT17a], but becomes possible once arithmetic reasoning is available [Sim17, BT17b].

Cyclic arithmetic was proposed as a general subject of study by Simpson in [Sim17]. Working in the language of arithmetic, it replaces induction by non-wellfounded proofs with a certain 'fairness' condition on the infinite branches. The advantage of this approach to infinite proof theory as opposed to, say, infinite well-founded proofs via an ω-rule (see, e.g., [Sch77]), is that it admits a notion of *finite proofs*: those that have only finitely many distinct subproofs, and so may be represented by a finite (possibly cyclic) graph.

Cyclic arithmetic itself is to cyclic proofs what Peano arithmetic is to traditional proofs: it provides a general framework in which many arguments can be interpreted and/or proved in a uniform manner, and this is one reason why it is an interesting subject of study. This is already clear from, say, the results of [BT17b], where the study of cyclic proofs for first-order logic with inductive definitions relied on an underlying arithmetic framework. We elaborate further on this in Sect. 10.

Contribution. In [Sim17], Simpson showed that Peano Arithmetic (PA) is able to simulate cyclic reasoning by proving the soundness of the latter in the former. (The converse result is obtained much more easily.) Nonetheless, several open questions remain from [Sim17], concerning constructivity, normalisation, logical complexity and proof complexity for cyclic and non-wellfounded proofs.

In this work we address the *logical complexity* and *proof complexity* of proofs in Cyclic Arithmetic (CA), as compared to PA. Namely, we study how quantifier alternation of proofs in one system compares to that in the other, and furthermore how the size of proofs compare. Writing $\mathrm{C}\Sigma_n$ for (the logical consequences of) cyclic proofs containing only Σ_n formulae, we show, for $n \geq 0$:

(1) $\mathrm{I}\Sigma_{n+1} \subseteq \mathrm{C}\Sigma_n$ over Π_{n+1} theorems (Sect. 4, Thm. 4.1).
(2) CA and PA proofs of the same theorem differ only exponentially in size (Sect. 6, Thm. 6.10).
(3) $\mathrm{C}\Sigma_n \subseteq \mathrm{I}\Sigma_{n+1}$ over all theorems (Sect. 7, Thm. 7.3).

(1) is obtained by proof theoretic techniques, relying on normal forms and structural manipulations of Peano Arithmetic proofs. It improves upon the natural result that $\mathrm{I}\Sigma_n \subseteq \mathrm{C}\Sigma_n$, although induces a non-elementary blowup in the size of proofs. (2) is obtained via a certain 'uniformisation' of the approach of [Sim17]. In particular, by specialising the key intermediate results to the case of cyclic proofs, we are able to extract small PA proofs of some required properties of infinite word automata from analogous ones in 'second-order' (SO) arithmetic. Finally, (3) is obtained by calibrating the argument of (2) with recent results on the reverse mathematics of Büchi's theorem [KMPS19], allowing us to bound the logical complexity of proofs in the simulation. Together, these results almost completely characterise the logical and proof complexity theoretic strength of cyclic proofs in arithmetic, answering the questions (ii) and (iii), Sect. 7 of [Sim17].

After demonstrating these results, we give a metamathematical analysis of provability in cyclic theories, in particular showing that the consistency of $\mathrm{C}\Sigma_n$ is provable in $\mathrm{I}\Sigma_{n+2}$ but not $\mathrm{I}\Sigma_{n+1}$, by appealing to a form of *Gödel incompleteness* for cyclic theories. We also use these observations to show that certain formulations of McNaughton's theorem, that every nondeterministic Büchi automaton has an equivalent deterministic parity (or Rabin,

Muller, etc.) automaton, are not provable in the SO theory RCA_0. This partially resolves the question of the logical strength of McNaughton's theorem left open in [KMPS19].

Structure of the paper. In Sects. 2 and 3 we introduce some preliminaries on Peano Arithmetic, proof theory, cyclic proofs and automaton theory. In Sect. 4 we present (1) and give an example of the translation in App. A. The contents of Sects. 2, 3 and 4 and App. A are more-or-less self contained and should be accessible to the general proof theorist.

We briefly introduce some SO theories of arithmetic in Sect. 5 that are conservative over the fragments of PA we need in order to conduct some of the intermediate arguments on infinite word automata. In Sect. 6 we present (2), and in Sect. 7 we adapt the argument to obtain (3). In Sect. 8 we give our metamathematical analysis of cyclic theories, and in Sect. 9 we explain their consequences for the logical strength of certain forms of McNaughton's theorem. We conclude with some further remarks and perspectives in Sect. 10, including a comparison with the results of [Sim17] and [BT17b].

For Sects. 5, 6, 7 and 9 it would be helpful for the reader to have some background in subsystems of second-order arithmetic (see e.g. [Sim09, Hir14]) and ω-automaton theory (see e.g. [Tho97]). For Sect. 8 it would be helpful for the reader to have some background in the metamathematics of first-order arithmetic (see e.g. [HP93]). Nonetheless we aim to give sufficient details for the general proof theorist to appreciate all the content herein.

2. Preliminaries on first-order arithmetic proof theory

We present only brief preliminaries, but the reader is encouraged to consult, e.g., [Bus98] for a more thorough introduction to first-order arithmetic.

We work in first-order (FO) logic with equality, $=$, with variables written x, y, z etc., terms written s, t, u etc., and formulae written φ, ψ etc., construed over the logical basis $\{\neg, \vee, \wedge, \exists, \forall\}$. We will usually assume formulae are in **De Morgan normal form**, with negation restricted to atomic formulae. Nonetheless, we may write $\neg\varphi$ for the De Morgan 'dual' of φ, defined as follows:

$$\neg\neg\varphi := \varphi \qquad \begin{array}{lcllcl} \neg(\varphi \wedge \psi) & := & \neg\varphi \vee \neg\psi & \neg\forall x.\varphi & := & \exists x.\neg\varphi \\ \neg(\varphi \vee \psi) & := & \neg\varphi \wedge \neg\psi & \neg\exists x.\varphi & := & \forall x.\neg\varphi \end{array}$$

We also write $\varphi \supset \psi$ for $\neg\varphi \vee \psi$ and $\varphi \equiv \psi$ for $(\varphi \supset \psi) \wedge (\psi \supset \varphi)$.

FO logic has equality 'built-in', i.e. we always assume the following axioms are present:

(eq1) $\forall x.x = x$.

(eq2) $\forall \vec{x}, \vec{y}.((x_1 = y_1 \wedge \cdots \wedge x_k = y_k) \supset f(\vec{x}) = f(\vec{y}))$, for each $k \in \mathbb{N}$ and each function symbol f of arity k.

(eq3) $\forall x, y.(((x_1 = y_1 \wedge \cdots \wedge x_k = y_k) \wedge P(\vec{x})) \supset P(\vec{y}))$, for each $k \in \mathbb{N}$ and each predicate symbol P of arity k.

Following [Sim17], the **language of arithmetic** (with inequality) is formulated as $\{0, \mathsf{s}, +, \times, <\}$, with their usual interpretations over $\mathbb{N}$. A **theory** is a set T of closed formulae over this language. We write $T \vdash \varphi$ if φ is a logical consequence of T. We write $T_1 \subseteq T_2$ if $T_1 \vdash \varphi$ implies $T_2 \vdash \varphi$, and $T_1 = T_2$ if $T_1 \subseteq T_2$ and $T_2 \subseteq T_1$.

The theory of **Robinson arithmetic** (with inequality), written Q, is axiomatised by:

(Q1) $\forall x.\mathsf{s}x \neq 0$.

(Q2) $\forall x, y.(\mathsf{s}x = \mathsf{s}y \supset x = y)$.

(Q3) $\forall x.(x \neq 0 \supset \exists y.x = \mathsf{s}y)$.

(Q4) $\forall x.\ x + 0 = x$.
(Q5) $\forall x, y.\ x + \mathsf{s}y = \mathsf{s}(x + y)$.
(Q6) $\forall x.\ x \cdot 0 = 0$.
(Q7) $\forall x, y.\ x \cdot \mathsf{s}y = x \cdot y + x$.
(Q8) $\forall x, y.(x < y \equiv \exists z.(x + \mathsf{s}z = y))$

Notice that, above and elsewhere, we may write $\cdot$ instead of $\times$ in terms, or even omit the symbol altogether, and we assume it binds more strongly than $+$. We also write $\forall x < t.\varphi$ and $\exists x < t.\varphi$ as abbreviations for $\forall x.(x < t \supset \varphi)$ and $\exists x.(x < t \wedge \varphi)$ resp. Formulae with only such quantifiers are called **bounded**.

As usual, we may assume that Q is axiomatised by the universal closures of bounded formulae. In particular the existential quantifiers in axioms (Q3) and (Q8) above may be bounded by x and y resp., provably under quantifier-free induction. We will implicitly assume this bounded axiomatisation for the sequent calculus formulation of arithmetic later.

Remark 2.1. Our basic axioms and, later, our inference rules differ slightly from those in [Sim17], however it is routine to see that the theories PA and CA defined in this work coincide with those of [Sim17]. In particular the axiomatisations are equivalent once even open induction is present (this is weaker than any theory we will consider). We chose a slightly different presentation so that we could readily apply certain metalogical results, such as Thm. 2.5, with no intermediate proof manipulation.

Definition 2.2 (Arithmetical hierarchy)**.** For $n \geq 0$, we define:

- $\Delta_0 = \Pi_0 = \Sigma_0$ is the class of bounded formulae.
- Σ_{n+1} is the class of formulae of the form $\exists \vec{x}.\varphi$, where $\varphi \in \Pi_n$.
- Π_{n+1} is the class of formulae of the form $\forall \vec{x}.\varphi$, where $\varphi \in \Sigma_n$.

Notice in particular that, by definition of De Morgan normal form, if $\varphi \in \Sigma_n$ then $\neg\varphi \in \Pi_n$ and vice-versa. In practice we often consider these classes of formulae up to logical equivalence. We say that a formula is in Δ_n (in a theory T) if it is equivalent (resp. provably equivalent in T) to both a Σ_n and Π_n formula.

Definition 2.3 (Arithmetic)**. Peano Arithmetic** (PA) is axiomatised by Q and the axiom schema of **induction**:

$$(\varphi(0) \wedge \forall x.(\varphi(x) \supset \varphi(\mathsf{s}x))) \supset \forall x.\varphi(x) \tag{2.1}$$

For a class of formulae Φ, we write Φ-IND for the set of induction axiom instances when $\varphi \in \Phi$ in (2.1). We write $\mathrm{I}\Phi$ for the theory $\mathsf{Q} + \Phi$-IND.

The following is a classical result:

Proposition 2.4 (See e.g. [Bus98, Kay91])**.** *For $n \geq 0$, we have $\mathrm{I}\Sigma_n = \mathrm{I}\Pi_n$.*

2.1. **A sequent calculus presentation of PA.** We will work with a standard sequent calculus presentation of FO logic, given in Fig. 1, where $i \in \{0, 1\}$ and a, known as the 'eigenvariable', is fresh, i.e. does not occur free in the lower sequent. Two important considerations are that we work with cedents as *sets*, i.e. there is no explicit need for contraction rules, and that we have an explicit **substitution** rule. In the *θ-sub* rule the 'substitution' θ is a mapping from variables to terms, which is extended in the natural way

$$\textit{id}\,\frac{}{\Gamma, \varphi \Rightarrow \Delta, \varphi} \qquad =_1 \frac{}{\Gamma \Rightarrow \Delta, t = t} \qquad =_2 \frac{}{\Gamma, s_1 = t_1, \ldots, s_k = t_k \Rightarrow \Delta, f(\vec{s}) = f(\vec{t})}$$

$$\neg\text{-}l\,\frac{}{\Gamma, \varphi, \neg\varphi \Rightarrow \Delta} \qquad \neg\text{-}r\,\frac{}{\Gamma \Rightarrow \Delta, \varphi, \neg\varphi} \qquad =_3 \frac{}{\Gamma, s_1 = t_1, \ldots, s_k = t_k, P(\vec{s}) \Rightarrow \Delta, P(\vec{t})}$$

$$\theta\text{-}sub\,\frac{\Gamma \Rightarrow \Delta}{\theta(\Gamma) \Rightarrow \theta(\Delta)} \qquad cut\,\frac{\Gamma \Rightarrow \Delta, \varphi \quad \Gamma, \varphi \Rightarrow \Delta}{\Gamma \Rightarrow \Delta} \qquad wk\,\frac{\Gamma \Rightarrow \Delta}{\Gamma, \Gamma' \Rightarrow \Delta, \Delta'}$$

$$\vee\text{-}l\,\frac{\Gamma, \varphi \Rightarrow \Delta \quad \Gamma, \psi \Rightarrow \Delta}{\Gamma, \varphi \vee \psi \Rightarrow \Delta} \qquad \wedge\text{-}l\,\frac{\Gamma, \varphi_i \Rightarrow \Delta}{\Gamma, \varphi_0 \wedge \varphi_1 \Rightarrow \Delta} \qquad \exists\text{-}l\,\frac{\Gamma, \varphi[a/x] \Rightarrow \Delta}{\Gamma, \exists x.\varphi \Rightarrow \Delta} \qquad \forall\text{-}l\,\frac{\Gamma, \varphi[t/x] \Rightarrow \Delta}{\Gamma, \forall x.\varphi \Rightarrow \Delta}$$

$$\wedge\text{-}r\,\frac{\Gamma \Rightarrow \Delta, \varphi \quad \Gamma \Rightarrow \Delta, \psi}{\Gamma \Rightarrow \Delta, \varphi \wedge \psi} \qquad \vee\text{-}r\,\frac{\Gamma \Rightarrow \Delta, \varphi_i}{\Gamma \Rightarrow \Delta, \varphi_0 \vee \varphi_1} \qquad \forall\text{-}r\,\frac{\Gamma \Rightarrow \Delta, \varphi[a/x]}{\Gamma \Rightarrow \Delta, \forall x.\varphi} \qquad \exists\text{-}r\,\frac{\Gamma \Rightarrow \Delta, \varphi[t/x]}{\Gamma \Rightarrow \Delta, \exists x.\varphi}$$

Figure 1: The sequent calculus for FO logic with equality, where a occurs only as indicated and $i \in \{0, 1\}$.

to cedents. Substitution is important for the definition of a cyclic arithmetic proof in the next section, but does not change provability in usual proofs.

The sequent calculus for Q is obtained from the FO calculus in the language of arithmetic by adding appropriate initial sequents for each instantiation of an axiom of Q by terms. For theories extending Q by (at least quantifier-free) induction, we assume that these intial sequents contain only Δ_0 formulae by appropriately bounding the existential quantifiers. The schema Φ-IND, for Φ closed under subformulas and substitution, is implemented in the calculus by adding the induction rule,

$$\textit{ind}\,\frac{\Gamma \Rightarrow \varphi(0), \Delta \quad \Gamma, \varphi(a) \Rightarrow \varphi(\mathsf{s}a), \Delta}{\Gamma \Rightarrow \varphi(t), \Delta}$$

for formulae $\varphi \in \Phi$. Here we require a to not occur free in the lower sequent. Notice that this satisfies the subformula property, in the 'wide' sense of FO logic, i.e. up to substitution. For fragments of PA with induction axioms of bounded logical complexity, we also have the *bounded quantifier rules*:

$$\frac{\Gamma, a < s, \varphi(a) \Rightarrow \Delta}{\Gamma, \exists x < s.\varphi(x) \Rightarrow \Delta} \qquad \frac{\Gamma \Rightarrow \Delta, \varphi(t)}{\Gamma, t < s \Rightarrow \Delta, \exists x < s.\varphi(x)}$$

$$\frac{\Gamma, a < s \Rightarrow \Delta, \varphi(a)}{\Gamma \Rightarrow \Delta, \forall x < s.\varphi(x)} \qquad \frac{\Gamma, \varphi(t) \Rightarrow \Delta}{\Gamma, t < s, \forall x < s.\varphi(x) \Rightarrow \Delta}$$

In all cases the eigenvariable a occurs only as indicated,

The following normalisation result is well-known in the proof theory of arithmetic, and will be one of the main structural proof theoretic tools in this work:

Theorem 2.5 (Free-cut elimination, e.g. [Bus98])**.** *Let $\mathcal{S}$ be a sequent system extending FO by the induction rule and some other nonlogical rules/axioms closed under substitution. Then any $\mathcal{S}$-proof can be effectively transformed into one of the same conclusion containing only (substitution instances of) subformulae of the conclusion, an induction formula or a formula occurring in another nonlogical step.*

Naturally, this applies to the various fragments of PA that we consider. In particular, notice that a free-cut free proof in $\mathrm{I}\Sigma_n$ or $\mathrm{I}\Pi_n$ of sequents containing only Σ_n or Π_n formulae, resp., contains just Σ_n or Π_n formulae, resp. It is well known that Thm. 2.5 can itself be proved within $\mathrm{I}\Sigma_1$ and even weaker theories (see, e.g., [HP93]), under an appropriate coding of mathematical objects. We use this observation later in Sect. 8.

We say that a sequent is Σ_n (or Π_n) if it contains only Σ_n (resp. Π_n) formulae. A slight issue that will be relevant later in Sect. 4 is that we have not defined Σ_n and Π_n as being syntactically closed under positive Boolean combinations, even if semantically we know that they are. In fact, this does not cause a problem for the result above, since we can always prenex 'on the fly' in a proof by cutting against appropriate derivations. For instance, in a proof, a step of the form,

$$\wedge \frac{\Gamma \Rightarrow \Delta, \forall x.\varphi \quad \Gamma \Rightarrow \Delta, \forall y.\psi}{\Gamma \Rightarrow \Delta, \forall x.\varphi \wedge \forall y.\psi}$$

may be locally replaced by a derivation of the form:

$$cut \frac{\Gamma \Rightarrow \Delta, \forall x.\varphi \qquad cut \dfrac{\Gamma \Rightarrow \Delta, \forall y.\psi \quad \dfrac{}{\forall x.\varphi, \forall y.\psi \Rightarrow \forall x, y.(\varphi \wedge \psi)}}{\Gamma, \forall y.\psi \Rightarrow \Delta, \forall x, y.(\varphi \wedge \psi)}}{\Gamma \Rightarrow \Delta, \forall x, y.(\varphi \wedge \psi)}$$

In a similar way we will often assume that a 'block' of existential or universal quantifiers is coded by a single quantifier, using pairings and Gödel β functions, whose basic properties are all formalisable already in $\mathrm{I}\Delta_0$ (see, e.g., [Bus98]).

3. Preliminaries on cyclic arithmetic and automata

Before presenting 'cyclic arithmetic', we will present the general notion of non-wellfounded proofs in arithmetic, from [Sim17].

By convention, we say *binary tree* to mean a nonempty set $T \subseteq \{0,1\}^*$ that is prefix-closed, i.e. if $\sigma i \in T$ then $\sigma \in T$. We construe such T as a bona fide tree with nodes T and directed edges from σ to σi, if $\sigma i \in T$, for $i \in \{0,1\}$. The empty word, ε, is the *root* of T.

Definition 3.1. A **preproof** is a possibly infinite binary tree labelled by sequents and rules in a locally correct manner in the calculus for Q. Following [Sim17], we treat inference steps as nodes of the tree and sequents as edges. A preproof is **regular** if it has only finitely many distinct (labelled) subtrees or, equivalently, if it is the unfolding of a finite labelled directed graph, possibly with cycles.

The following notions are variants of those from Dfns. 1 and 2 in [Sim17]:

Definition 3.2 (Precursors, traces, ∞-proofs)**.** Let $(\Gamma_i \Rightarrow \Delta_i)_{i \geq 0}$ be an infinite branch through a preproof. For terms t, t' we say that t' is a **precursor** of t at i if one of the following holds:

i) $\Gamma_i \Rightarrow \Delta_i$ concludes a θ-*sub*-step and t is $\theta(t')$.
ii) $\Gamma_i \Rightarrow \Delta_i$ concludes any other step and $t' = t$ occurs in Γ_i.
iii) $\Gamma_i \Rightarrow \Delta_i$ concludes any other step and t' is t.

A **trace** along $(\Gamma_i \Rightarrow \Delta_i)_{i \geq 0}$ is a sequence $(t_i)_{i \geq n}$, for some $n \geq 0$, such that whenever $i \geq n$ the term t_i occurs in $\Gamma_i \Rightarrow \Delta_i$ and,

(a) t_{i+1} is a precursor of t_i at i; or
(b) the atomic formula $t_{i+1} < t$ occurs in Γ_{i+1}, where t is a precursor of t_i at i.

When (b) holds, we say that the trace **progresses** at $i+1$.

An **∞-proof** is a preproof for which any infinite branch has a trace that progresses infinitely often. If it is regular then we simply call it a **cyclic proof**. CA is the theory induced by cyclic proofs in the calculus for Q.

Remark 3.3. When defining explicit traces, for the precursor case iii), we will typically not worry about whether the term t_i in a trace occurs in the sequent or not. All that matters is that, if the current step is $\exists$-l, $\forall$-r or *sub*, t_i does not contain the associated eigenvariables.[1] As long as we satisfy this constraint we may simply consider an equivalent proof that prepends $t_i = t_i$ to the antecedent to make sure that t_i 'occurs'. We use this assumption implicitly in the remainder of this work.

The reader may consult [Sim17] for several examples of ∞-proofs. Notably, ∞-proofs are sound and complete for the standard model $\mathbb{N}$ (Thm. 4, [Sim17]). (Similar results for other logics, with respect to standard models, were known before [Bro06, BS11].) We recall the proof of soundness since we will have to formalise a variant of it in Sect. 6, and also since the quantifier case in the argument of [Sim17] is omitted, whereas this subtlety will need some consideration when it is formalised.

Proposition 3.4 (Soundness of ∞-proofs)**.** *If π is an ∞-proof of φ, then $\mathbb{N} \vDash \varphi$.*

Proof. Suppose otherwise, i.e. $\mathbb{N} \vDash \neg\varphi$. We will inductively construct an infinite branch $(\Gamma_i \Rightarrow \Delta_i)_{i\geq 0}$ of π and associated assignments ρ_i of natural numbers to each sequent's free variables, such that $\mathbb{N}, \rho_i \nvDash \Gamma_i \Rightarrow \Delta_i$. Assuming φ is closed (by taking its universal closure), we set $\Gamma_0 \Rightarrow \Delta_0$ to be $\Rightarrow \varphi$ and $\rho_0 = \varnothing$.

Each step except for substitution, $\forall$-r and $\exists$-l constitutes a true implication, so if $\mathbb{N}, \rho_i \nvDash \Gamma_i \Rightarrow \Delta_i$ then ρ_i also must not satisfy one of its premisses. We may thus choose one such premiss as $\Gamma_{i+1} \Rightarrow \Delta_{i+1}$ and set $\rho_{i+1} = \rho_i$.

If $\Gamma_i \Rightarrow \Delta_i$ concludes a θ-*sub* step, we may set $\rho_{i+1} = \rho_i \circ \theta$. If $\Gamma_i \Rightarrow \Delta_i$ concludes a $\forall$-r step, let $\forall x.\varphi$ be the principal formula and assume x does not occur free in the conclusion. Since $\mathbb{N}, \rho_i \nvDash \Gamma_i \Rightarrow \Delta_i$, we must have that $\mathbb{N}, \rho_i \vDash \exists x.\neg\varphi$. We choose a value $k \in \mathbb{N}$ witnessing this existential and set $\rho_{i+1} = \rho_i \cup \{x \mapsto k\}$. The $\exists$-l case is dealt with similarly.

This infinite branch must have an infinitely progressing trace, say $(t_i)_{i\geq n}$, by the definition of ∞-proof. However notice that, for $i \geq n$, $\rho_i(t_i) \geq \rho_{i+1}(t_{i+1})$ and, furthermore, at a progress point along the trace, $\rho_i(t_i) > \rho_{i+1}(t_{i+1})$. Thus, $(\rho_i(t_i))_{i\geq n}$ is a monotone decreasing sequence of natural numbers that does not converge, contradicting the fact that $\mathbb{N}$ is well-ordered. □

Later, in Sect. 6, we will use the fact that the choices for generating an invalid branch in the proof above can be made *uniformly* in an arithmetic setting.

3.1. **Defining $\mathrm{C}\Sigma_n$.** Simpson proposes in [Sim17] to study systems of cyclic proofs containing only Σ_n formulae, and to compare such systems to $\mathrm{I}\Sigma_n$. This is rather pertinent in light of the free-cut elimination result we stated, Thm. 2.5: any $\mathrm{I}\Sigma_n$-proof of a Σ_n-sequent can be assumed to contain just Σ_n formulae (possibly at a non-elementary cost in proof size), whence the comparison. However, in order to be able to admit routine derivations of more

[1] We say that a is an eigenvariable of a θ-*sub* step if it is in the support of the substitution θ.

complex formulae, e.g. the Σ_{n+1} law of excluded middle or the universal closure of a Σ_n sequent, we will close this notion under *logical consequence*.

Definition 3.5. Let Φ be a set of formulae closed under subformulae and substitution. $\mathrm{C}\Phi$ is the first-order theory axiomatised by the universal closures of conclusions of cyclic proofs containing only Φ-formulae.

Notice that, by the free-cut elimination result, Thm. 2.5, and the subformula property, any $\mathrm{C}\Sigma_n$ proofs of Σ_n-sequents contain only Σ_n-sequents anyway, without loss of generality. This more 'robust' definition allows us to easily compare fragments of cyclic arithmetic. For instance, we have the following:

Proposition 3.6. $\mathrm{C}\Sigma_n = \mathrm{C}\Pi_n$, *for* $n \geq 0$.

Proof. For the left-right inclusion, replace each Σ_n sequent $\vec{p}, \Gamma \Rightarrow \Delta$ with the sequent $\vec{p}, \overline{\Delta} \Rightarrow \overline{\Gamma}$, where $\overline{\Gamma}$ and $\overline{\Delta}$ contain the De Morgan dual formulae of Γ and Δ resp. and $\vec{p}$ exhausts the atomic formulae of the antecedent. Any traces will be preserved and the proof can be made correct by locally adding some logical steps. The converse implication is proved in the same way. □

Using a standard technique, e.g. from [Bro06], we also can rather simply show the following result, which we will later strengthen in Sect. 4:[2]

Proposition 3.7. $\mathrm{C}\Sigma_n$ *proves any* Π_{n+1} *theorem of* $\mathrm{I}\Sigma_n$, *for* $n \geq 0$.

Proof sketch. Suppose $\mathrm{I}\Sigma_n$ proves $\forall \vec{x}.\psi(\vec{x})$ where φ is Σ_n. Let π be a free-cut free $\mathrm{I}\Sigma_n$ proof of $\psi(\vec{a})$, so in particular contains only Σ_n formulas, by the subformula property. We may now construct a $\mathrm{C}\Sigma_n$ proof of $\varphi(\vec{a})$ by simply simulating every local inference step of π; the only nontrivial case is the induction rule:

$$\mathit{ind}\ \frac{\Gamma \Rightarrow \varphi(0), \Delta \quad \Gamma, \varphi(a) \Rightarrow \varphi(\mathsf{s}a), \Delta}{\Gamma \Rightarrow \varphi(t), \Delta}$$

This is simulated by the following cyclic derivation (omitting some routine proof steps),

$$\mathit{sub}\ \dfrac{\mathit{cut}\ \dfrac{\Gamma \Rightarrow \varphi(0), \Delta \qquad \mathit{cut}\ \dfrac{\mathit{sub}\ \dfrac{\vdots}{\Gamma \Rightarrow \varphi(b), \Delta}\bullet}{\Gamma \Rightarrow \varphi(a), \Delta} \qquad \Gamma, \varphi(a) \Rightarrow \varphi(\mathsf{s}a), \Delta}{0 = b, \Gamma \Rightarrow \varphi(b), \Delta} \qquad \dfrac{\dfrac{\dfrac{\underline{a < b}, \Gamma \Rightarrow \varphi(\mathsf{s}a), \Delta}{b = \mathsf{s}a, \Gamma \Rightarrow \varphi(b), \Delta}}{0 < b, \Gamma \Rightarrow \varphi(b), \Delta}}{\Gamma \Rightarrow \varphi(b), \Delta}\bullet}{\Gamma \Rightarrow \varphi(t), \Delta}$$

where we have written $\bullet$ to mark roots of identical subtrees. An infinite branch that does not have a tail in the proofs of the two premisses of *ind* must eventually loop on $\bullet$. Therefore it admits an infinitely progressing trace alternating between a and b, with the progress point underlined above. Now the proposition follows by simple application of $\forall$-r. □

[2] A similar result was given in [Sim17], but that argument rather shows that $\mathrm{I}\Sigma_n \subseteq \mathrm{C}\Sigma_{n+1}$.

Following Rmk. 3.3, notice that, e.g. in the simulation of induction above, traces need not be connected in the graph of ancestry of a proof. This deviates from other settings where it is *occurrences* that are tracked, rather than terms, e.g. in [DBHS16, BDS16, Dou17].

3.2. **Büchi automata: checking correctness of cyclic proofs.** A cyclic preproof can be effectively checked for correctness by reduction to the inclusion of 'Büchi automata', yielding a **PSPACE** bound. As far as the author is aware, this is the best known upper bound, although no corresponding lower bound is known. As we will see later in Sect. 6, this is one of the reasons why we cannot hope for a 'polynomial simulation' of cyclic proofs in a usual proof system, and so why elementary simulations are more pertinent.

Definition 3.8. A **nondeterministic Büchi automaton** (NBA) $\mathcal{A}$ is a tuple (A, Q, δ, q_0, F) where: A is a finite set, called the **alphabet**, Q is a finite set of **states**, $\delta \subseteq (Q \times A) \times Q$ is the **transition relation**, $q_0 \in Q$ is the **initial** state, and $F \subseteq Q$ is the set of **final** or **accepting** states. We say that $\mathcal{A}$ is **deterministic** (a DBA) if δ is (the graph of) a function $Q \times A \to Q$. A 'word' $(a_i)_{i\geq 0} \in A^\omega$ is **accepted** or **recognised** by $\mathcal{A}$ if there is a sequence $(q_i)_{i\geq 0} \in Q^\omega$ such that: for each $i \geq 0$, $(q_i, a_i, q_{i+1}) \in \delta$, and for infinitely many $i \geq 0$ we have $q_i \in F$. We write $\mathcal{L}(\mathcal{A})$ for the set of words in A^ω accepted by $\mathcal{A}$.

From a cyclic preproof π we can easily define two automata, say $\mathcal{A}^\pi_b$ and $\mathcal{A}^\pi_t$,[3] respectively accepting just the infinite branches and just the infinite branches with infinitely progressing traces. See [Sim17] for a construction of $\mathcal{A}^\pi_t$. We point out that $\mathcal{A}^\pi_b$ is essentially just the dependency graph of π with all states final, and so is in fact deterministic;[4] we will rely on this observation later in Sects. 6, 7 and 9. We now state the well-known 'correctness criterion' for cyclic proofs:

Proposition 3.9 [Sim17]**.** *A cyclic preproof π is a ∞-proof iff $\mathcal{L}(\mathcal{A}^\pi_b) \subseteq \mathcal{L}(\mathcal{A}^\pi_t)$.*

4. A translation from $\mathrm{I}\Sigma_{n+1}$ to $\mathrm{C}\Sigma_n$, over Π_{n+1}-theorems

We show in this section our first result, that cyclic proofs containing only Σ_n-formulae are enough to simulate $\mathrm{I}\Sigma_{n+1}$ over not-too-complex formulae:

Theorem 4.1. *$\mathrm{I}\Sigma_{n+1} \subseteq \mathrm{C}\Sigma_n$, over Π_{n+1} theorems, for $n \geq 0$.*

One example of such logical power in cyclic proofs was given in [Sim17], in the form of $\mathrm{C}\Sigma_1$ proofs of the totality of the Ackermann-Péter function. This already separates it from $\mathrm{I}\Sigma_1$, which only proves the totality of the primitive recursive functions [Par72]. To prove the theorem above, we will rather work in III_{n+1}, cf. Prop. 2.4, since the exposition is more intuitive. We first prove the following intermediate lemma.

Lemma 4.2. *Let π be a III_{n+1} proof, containing only Π_{n+1} formulae, of a sequent,*

$$\Gamma, \forall x_1.\varphi_1, \dots, \forall x_l.\varphi_l \Rightarrow \Delta, \forall y_1.\psi_1, \dots, \forall y_m.\psi_m \tag{4.1}$$

[3] These are rather called B_p and B_t respectively in [Sim17].

[4] Technically the transition relation here is not total, but this can be 'completed' in the usual way by adding a non-final 'sink' state for any outstanding transitions.

where $\Gamma, \Delta, \varphi_i, \psi_j$ *are* Σ_n *and* x_i, y_j *occur only in* φ_i, ψ_j *respectively. Then there is a* $\mathrm{C}\Sigma_n$ *derivation* $\lceil\pi\rceil$ *of the form:*

$$\dfrac{\{\Gamma \Rightarrow \Delta, \varphi_i\}_{i\leq l}}{\Gamma \Rightarrow \Delta, \psi_1, \ldots, \psi_m}\ \lceil\pi\rceil$$

Moreover, no free variables of (4.1) *occur as eigenvariables for* $\exists$*-*l*,* $\forall$*-*r *or sub steps in* $\lceil\pi\rceil$*.*

Proof. We proceed by induction on the structure of π. Notice that we may assume that any Π_{n+1} formulae occurring have just a single outermost $\forall$ quantifier, by interpreting arguments as pairs and using Gödel's β functions. (This introduces only cuts on formulae of the same form.) We henceforth write $\vec{\varphi}$ for $\varphi_1, \ldots, \varphi_l$ and $\vec{\psi}$ for $\psi_1, \ldots, \psi_m$ and, as an abuse of notation, $\forall\vec{x}.\vec{\varphi}$ and $\forall\vec{y}.\vec{\psi}$ for $\forall x_1.\varphi_1, \ldots, \forall x_l.\varphi_l$ and $\forall y_1.\psi_1, \ldots, \forall y_m.\psi_m$ respectively. (Notice that this is a reasonable abuse of notation, since $\forall$s can be prenexed outside conjunctions and disjunctions already in pure FO logic.)

Propositional logical steps are easily dealt with, relying on invertibility and cuts, with possible structural steps. Importantly, due to the statement of the lemma, such steps apply to only Σ_n formulae (recall the discussion at the end of Sect. 2). For instance, if π extends a proof π' by a $\wedge$-left step,

$$\wedge\text{-}l\ \dfrac{\Gamma, \chi_0, \chi_1, \forall\vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall\vec{y}.\vec{\psi}}{\Gamma, \chi_0 \wedge \chi_1, \forall\vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall\vec{y}.\vec{\psi}}$$

then we define $\lceil\pi\rceil$ as,

$$\dfrac{\left\{ cut\ \dfrac{\dfrac{}{\chi_0, \chi_1 \Rightarrow \chi_0 \wedge \chi_1} \quad \Gamma, \chi_0 \wedge \chi_1 \Rightarrow \Delta, \varphi_i}{\Gamma, \chi_0, \chi_1 \Rightarrow \Delta, \varphi_i} \right\}_{i\leq l}}{\wedge\text{-}l\ \dfrac{\Gamma, \chi_0, \chi_1 \Rightarrow \Delta, \vec{\psi}}{\Gamma, \chi_0 \wedge \chi_1 \Rightarrow \Delta, \vec{\psi}}}\ \lceil\pi'\rceil$$

and if π extends proofs π_0 and π_1 by a $\wedge$-right step,

$$\wedge\text{-}r\ \dfrac{\Gamma, \forall\vec{x}.\vec{\varphi} \Rightarrow \Delta, \chi_0, \forall\vec{y}.\vec{\psi} \quad \Gamma, \forall\vec{x}.\vec{\varphi} \Rightarrow \Delta, \chi_1, \forall\vec{y}.\vec{\psi}}{\Gamma, \forall\vec{x}.\vec{\varphi} \Rightarrow \Delta, \chi_0 \wedge \chi_1, \forall\vec{y}.\vec{\psi}}$$

then we define $\lceil\pi\rceil$ as:

$$\dfrac{\left\{ cut\ \dfrac{\Gamma \Rightarrow \Delta, \chi_0 \wedge \chi_1, \varphi_i \quad \dfrac{}{\chi_0 \wedge \chi_1 \Rightarrow \chi_j}}{\Gamma \Rightarrow \Delta, \chi_j, \varphi_i} \right\}_{i\leq l}}{\wedge\text{-}r\ \dfrac{\Gamma \Rightarrow \Delta, \chi_j, \vec{\psi}}{\Gamma \Rightarrow \Delta, \chi_0 \wedge \chi_1, \vec{\psi}}\ \forall j < 2}\ \lceil\pi_j\rceil$$

If π extends a proof π' by a thinning step,

$$wk\,\frac{\Gamma, \forall \vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall \vec{y}.\vec{\psi}}{\Gamma', \Pi, \Gamma, \forall \vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall \vec{y}.\vec{\psi}, \Delta', \forall \vec{z}.\vec{\chi}}$$

where $\Gamma', \Delta', \vec{\chi}$ are Σ_n and Π is Π_{n+1}, then we define $\lceil \pi \rceil$ as:

$$wk\,\frac{\dfrac{\{\Gamma', \Gamma \Rightarrow \Delta, \varphi_i, \Delta'\}_{i \leq l}}{\Gamma', \lceil \pi' \rceil, \Delta'}}{\Gamma', \Gamma \Rightarrow \Delta, \vec{\psi}, \Delta', \vec{\chi}}$$

where $\Gamma', \lceil \pi' \rceil, \Delta'$ is obtained from $\lceil \pi' \rceil$ by prepending Γ' and appending Δ' to each sequent. For this we might need to rename some free variables in π' so that eigenvariable conditions are preserved after the transformation; this does not affect the cedents Γ, Δ by the assumption from the inductive hypothesis. Notice that we are simply ignoring the extra premisses due to Π.

If π extends proofs π_0 and π_1 by a *cut* step on a Π_{n+1} formula,

$$cut\,\frac{\Gamma, \forall \vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall \vec{y}.\vec{\psi}, \forall z.\chi \quad \Gamma, \forall \vec{x}.\vec{\varphi}, \forall z.\chi \Rightarrow \Delta, \forall \vec{y}.\vec{\psi}}{\Gamma, \forall \vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall \vec{y}.\vec{\psi}}$$

then we define $\lceil \pi \rceil$ as:

$$\dfrac{\dfrac{\dfrac{\{\Gamma \Rightarrow \Delta, \varphi_i\}_{i \leq l}}{\Gamma \Rightarrow \Delta, \vec{\psi}, \chi} \quad \left\{ wk\,\dfrac{\Gamma \Rightarrow \Delta, \varphi_i}{\Gamma \Rightarrow \Delta, \vec{\psi}, \varphi_i} \right\}_{i \leq l}}{\Gamma \Rightarrow \Delta, \vec{\psi}, \vec{\psi}}}{\Gamma \Rightarrow \Delta, \vec{\psi}}$$

The final dotted 'contraction' step is implicit, since we treat cedents as sets. Again, we might need to rename some variables in π_1. If instead the cut formula were Σ_n, say χ, we would define $\lceil \pi \rceil$ as:

$$cut\,\frac{\dfrac{\{\Gamma \Rightarrow \Delta, \varphi_i\}_{i \leq l}}{\Gamma \Rightarrow \Delta, \chi, \vec{\psi}} \quad \dfrac{\left\{ wk\,\dfrac{\Gamma \Rightarrow \Delta, \varphi_i}{\Gamma, \chi \Rightarrow \Delta, \varphi_i} \right\}_{i \leq l}}{\Gamma, \chi \Rightarrow \Delta, \vec{\psi}}}{\Gamma \Rightarrow \Delta, \vec{\psi}}$$

If π extends a proof π' by a $\forall$-left step,

$$\forall\text{-}l\,\frac{\Gamma, \chi(t), \forall\vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall\vec{y}.\vec{\psi}}{\Gamma, \forall z.\chi(z), \forall\vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall\vec{y}.\vec{\psi}}$$

where $\forall z.\chi(z)$ is Π_{n+1}, we define $\lceil\pi\rceil$ as follows:

$$\dfrac{\dfrac{\Gamma \Rightarrow \Delta, \chi(z)}{\Gamma \Rightarrow \Delta, \chi(t)}\,sub \qquad \begin{array}{c}\left\{wk\,\dfrac{\Gamma \Rightarrow \Delta, \varphi_i}{\Gamma, \chi(t) \Rightarrow \Delta, \varphi_i}\right\}_{i\leq l} \\ \hline \lceil\pi'\rceil \\ \hline \Gamma, \chi(t) \Rightarrow \Delta, \vec{\psi}\end{array}}{\Gamma \Rightarrow \Delta, \vec{\psi}}\,cut$$

(Notice that, although z occurs as an eigenvariable for a *sub* step here, it is already bound in the conclusion of π, so we preserve the inductive hypothesis.) If π extends a proof π' by a $\forall$-right step,

$$\forall\text{-}r\,\frac{\Gamma, \forall\vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall\vec{y}.\vec{\psi}, \chi}{\Gamma, \forall\vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall\vec{y}.\vec{\psi}, \forall z.\chi}$$

where $\forall z.\chi$ is Π_{n+1}, then we define $\lceil\pi\rceil$ as:

$$\begin{array}{c}\left\{wk\,\dfrac{\Gamma \Rightarrow \Delta, \varphi_i}{\Gamma \Rightarrow \Delta, \varphi_i, \chi}\right\}_{i\leq l} \\ \hline \lceil\pi'\rceil, \chi \\ \hline \Gamma \Rightarrow \Delta, \vec{\psi}, \chi\end{array}$$

If π extends a proof π' by a $\exists$-right step,

$$\exists\text{-}r\,\frac{\Gamma, \forall\vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall\vec{y}.\vec{\psi}, \chi(t)}{\Gamma, \forall\vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall\vec{y}.\vec{\psi}, \exists z.\chi(z)}$$

where $\exists z.\chi(z)$ is Σ_n, then we define $\lceil\pi\rceil$ as:

$$\exists\text{-}r\,\dfrac{\begin{array}{c}\left\{wk\,\dfrac{\Gamma \Rightarrow \Delta, \varphi_i, \exists z.\chi(z)}{\Gamma \Rightarrow \Delta, \varphi_i, \chi(t), \exists z.\chi(z)}\right\}_{i\leq l} \\ \hline \lceil\pi'\rceil, \exists z.\chi(z) \\ \hline \Gamma \Rightarrow \Delta, \vec{\psi}, \chi(t), \exists z.\chi(z)\end{array}}{\Gamma \Rightarrow \Delta, \vec{\psi}, \exists z.\chi(z)}$$

Again, some eigenvariables of π' might have to be renamed. Any other quantifier steps are dealt with routinely.

Finally, if π extends proofs π_0 and π' by an induction step,

$$ind\,\frac{\Gamma, \forall\vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall\vec{y}.\vec{\psi}, \forall z.\chi(0) \quad \Gamma, \forall\vec{x}.\vec{\varphi}, \forall z.\chi(c) \Rightarrow \Delta, \forall\vec{y}.\vec{\psi}, \forall z.\chi(\mathsf{s}c)}{\Gamma, \forall\vec{x}.\vec{\varphi} \Rightarrow \Delta, \forall\vec{y}.\vec{\psi}, \forall z.\chi(t)}$$

we define $\lceil \pi \rceil$ to be the following cyclic proof,

$$\dfrac{\dfrac{\{\Gamma \Rightarrow \Delta, \varphi_i\}_{i\leq l}}{\lceil \pi_0 \rceil}\quad \dfrac{\dfrac{\dfrac{\vdots}{\Gamma \Rightarrow \Delta, \vec{\psi}, \chi(d)}\bullet}{\Gamma \Rightarrow \Delta, \vec{\psi}, \chi(c)}\,sub \qquad \left\{\dfrac{\Gamma \Rightarrow \Delta, \varphi_i}{\Gamma \Rightarrow \Delta, \vec{\psi}, \varphi_i}\,wk\right\}_{i\leq l}}{\lceil \pi' \rceil, \vec{\psi}}}{\dfrac{\Gamma \Rightarrow \Delta, \vec{\psi}, \chi(0)}{d = 0, \Gamma \Rightarrow \Delta, \vec{\psi}, \chi(d)} \qquad \dfrac{\Gamma \Rightarrow \Delta, \vec{\psi}, \chi(\mathsf{s}c)}{\underline{c < d}, d = \mathsf{s}c, \Gamma \Rightarrow \Delta, \vec{\psi}, \chi(d)}}\bullet}{\Gamma \Rightarrow \Delta, \vec{\psi}, \chi(d)}\,sub$$

$$\Gamma \Rightarrow \Delta, \vec{\psi}, \chi(t)$$

where we have written $\bullet$ to mark roots of identical subtrees. Notice that any branch hitting $\bullet$ infinitely often will have an infinitely progressing trace alternating between c and d, by the underlined progress point $c < d$: thanks to the assumption from the inductive hypothesis, c will not occur in $\lceil \pi' \rceil$ as an eigenvariable for $\exists$-l, $\forall$-r or *sub* steps so the trace along c in $\lceil \pi' \rceil$ remains intact, cf. Rmk. 3.3. Any other infinite branch has a tail that is already in $\lceil \pi' \rceil$ or $\lceil \pi_0 \rceil$ and so has an infinitely progressing trace by the inductive hypothesis. □

The lemma above gives us a simple proof of the main result of this section:

Proof of Thm. 4.1. Let π be a III_{n+1} proof of a sequent $\Rightarrow \forall x.\varphi$, where $\varphi \in \Sigma_n$, under Prop. 2.4. By Thm. 2.5 we may assume that π contains only Π_{n+1} cuts, whence we may simply apply Lemma 4.2 to obtain a $\mathrm{C}\Sigma_n$ proof of $\Rightarrow \varphi$. (Notice that there are no assumption sequents after applying the lemma since the antecedent is empty.) Now the result follows simply by an application of $\forall$-r. □

For the interested reader, we have given an example of this translation in action in App. A, for a 'relativised' version of arithmetic with an uninterpreted function symbol.

5. Second-order theories for reasoning about automata

We now consider a two-sorted, or 'second-order' (SO), version of FO logic, with variables X, Y, Z, etc. ranging over sets of individuals, and new atomic formulae $t \in X$, sometimes written $X(t)$. We also have SO quantifiers binding the SO variables with the natural interpretation. Again, we give only brief preliminaries, but the reader is encouraged to consult the standard texts [Sim09] and [Hir14].

We write Q_2 for an appropriate extension of Q by basic axioms governing sets (see, e.g., [Sim09] or [Hir14]), and write Σ^0_n and Π^0_n for the classes Σ_n and Π_n respectively, but now allowing free set variables to occur.

Definition 5.1. The **recursive comprehension** axiom schema is the following:[5]

$$\Delta^0_1\text{-}\mathsf{CA} \;:\; \forall \vec{y}, \vec{Y}.(\forall x.(\varphi(x, \vec{y}, \vec{Y}) \equiv \neg\psi(x, \vec{y}, \vec{Y})) \supset \exists X.\forall x.(X(x) \equiv \varphi(x)))$$

[5]Notice that there is an unfortunate coincidence of the notation CA for 'comprehension axiom' and 'cyclic arithmetic', but the context of use should always avoid any ambiguity.

where φ, ψ are in Σ^0_1 and X does not occur free in φ or ψ. From here, the theory RCA_0 is defined as $\mathsf{Q}_2 + \Delta^0_1\text{-}\mathsf{CA} + \Sigma^0_1\text{-}\mathsf{IND}$.

Since we will always work in extensions of RCA_0, which proves the totality of primitive recursive functions, we will conservatively add function symbols for primitive recursive functions on individuals whenever we need them. We will also henceforth consider FO theories extended by 'oracles', i.e. uninterpreted set/predicate variables, in order to access 'uniform' classes of FO proofs. We write $\mathrm{I}\Sigma_n(X)$ for the same class of proofs as $\mathrm{I}\Sigma_n$ but where X is allowed to occur as a predicate symbol. The usefulness of a $\mathrm{I}\Sigma_n(X)$ proof is that we may later substitute X for a FO formula, say $\varphi(-) \in \Delta_{m+1}$, to arrive at a $\mathrm{I}\Sigma_{m+n}$ proof of size $O(|\varphi|)$. This 'parametrisation' of a FO proof allows us to avoid unnecessary blowups in proof size induced by 'non-uniform' translations from second-order theories; we implicitly use this observation for proof complexity bounds later, particularly in Sect. 6.

The following result is an adaptation of well known conservativity results, e.g. as found in [Sim09, Hir14], but we include a proof anyway for completeness.

Proposition 5.2. $\mathsf{RCA}_0 + \Sigma^0_n\text{-}\mathsf{IND}$ *is conservative over* $\mathrm{I}\Sigma_n(X)$.

Proof sketch. First we introduce countably many fresh set symbols $X^{\vec{t},\vec{Y}}_{\varphi,\psi}$, indexed by Σ^0_1 formulae $\varphi(x, \vec{x}, \vec{X}), \psi(x, \vec{x}, \vec{X})$ with all free variables indicated, FO terms $\vec{t}$ with $|\vec{t}| = |\vec{x}|$ and SO variables $\vec{Y}$ with $|\vec{Y}| = |\vec{X}|$. These will serve as witnesses to the sets defined by comprehension. We replace the comprehension axioms by initial sequents of the form:

$$\frac{}{\Gamma, \forall x.(\varphi(x, \vec{t}, \vec{Y}) \equiv \neg\psi(x, \vec{t}, \vec{Y})), \varphi(t, \vec{t}, \vec{Y}) \Rightarrow t \in X^{\vec{t},\vec{Y}}_{\varphi,\psi}, \Delta} \tag{5.1}$$

$$\frac{}{\Gamma, \forall x.(\varphi(x, \vec{t}, \vec{Y}) \equiv \neg\psi(x, \vec{t}, \vec{Y})), t \in X^{\vec{t},\vec{Y}}_{\varphi,\psi} \Rightarrow \varphi(t, \vec{t}, \vec{Y}), \Delta} \tag{5.2}$$

It is routine to show that these new initial sequents are equivalent to the comprehension axioms for φ, ψ.

Now we apply free-cut elimination, Thm. 2.5, to a proof in such a system and replace every occurrence of $t \in X^{\vec{t},\vec{Y}}_{\varphi,\psi}$ with $\varphi(t, \vec{t}, \vec{Y})$, and every occurrence of $t \notin X^{\vec{t},\vec{Y}}_{\varphi,\psi}$ with $\psi(t, \vec{t}, \vec{Y})$. (Recall here that we assume formulae are in De Morgan normal form.) Any comprehension initial sequents affected by this replacement become purely logical theorems. Furthermore, any induction formulae remain Σ^0_n, provably in pure logic, thanks to our consideration of whether $X^{\vec{t},\vec{Y}}_{\varphi,\psi}$ occurs positively or negatively. Any extraneous free set variables in induction steps (except X), e.g. Y, may be safely dealt with by replacing any atomic formula $Y(s)$ with $\top$. The resulting proof is in $\mathrm{I}\Sigma_n(X)$. □

It is worth pointing out that, in general, the transformation from a SO proof to a FO proof can yield a possibly non-elementary blowup in the size of proofs, due to, e.g., the application of (free-)cut elimination.

5.1. **Formalisation of Büchi acceptance.** From now on we will be rather informal when talking about finite objects, e.g. automata, finite sequences, or even formulae. In particular we may freely use such meta-level objects within object-level formulae when, in fact, we are formally referring to their 'Gödel numbers'. Also, statements inside quotations, "-", will usually be (provably) recursive in any free variables occurring, i.e. Δ^0_1. This way quantifier complexity is (usually) safely measured by just the quantifiers outside quotations.

We often treat a set symbol X as a binary predicate by interpreting its argument as a pair and using Gödel's 'β functions' to primitive-recursively extract its components. We use such predicates to encode sequences by interpreting $X(x, y)$ as "the x^{th} symbol of X is y"; this interpretation presumes we already have the totality and determinism of X as a binary relation. Formally, for a set S and a set symbol X treated as a binary predicate, we will write $X \in S^\omega$ for the conjunction of the following two formulae,

$$\forall x.\exists y \in S.X(x, y) \tag{5.3}$$

$$\forall x, y, z.((X(x, y) \wedge X(x, z)) \supset y = z) \tag{5.4}$$

i.e. X is, in fact, the graph of a function $\mathbb{N} \to S$. When we know that these formulae hold true for X, we may construe the expression $X(x)$ as a term in formulae, for instance writing $\varphi(X(x))$ as shorthand for $\exists y.(X(x, y) \wedge \varphi(y))$ or, equivalently, $\forall y.(X(x, y) \supset \varphi(y))$.

Definition 5.3 (Language membership)**.** Let $\mathcal{A} = (A, Q, \delta, q_0, F)$ be a NBA and treat X as a binary predicate symbol. We define the formula $X \in \mathcal{L}(\mathcal{A})$ as:

$$X \in A^\omega \ \wedge \ \exists Y \in Q^\omega. \left(\begin{array}{ll} & Y(0, q_0) \\ \wedge & \forall x.\ (Y(x), X(x), Y(\mathsf{s}x)) \in \delta \\ \wedge & \forall x.\exists x' > x.\ Y(x') \in F \end{array} \right) \tag{5.5}$$

If $\mathcal{A}$ is deterministic and $X \in A^\omega$, we write $q_X(x, y)$ for "y is the x^{th} state of the run of X on $\mathcal{A}$", which is provably recursive in RCA_0. Similarly to before, we may write $\varphi(q_X(x))$ as shorthand for $\exists y.(q_X(x, y) \wedge \varphi(y))$ or equivalently in RCA_0, for $\forall y.(q_X(x, y) \supset \varphi(y))$. For DBA, we alternatively define $X \in \mathcal{L}(\mathcal{A})$ as:

$$X \in A^\omega \ \wedge \ \forall x.\exists x' > x.\ q_X(x') \in F \tag{5.6}$$

This 'double definition' will not be problematic for us, since RCA_0 can check if an automaton is deterministic or not and, if so, even prove the equivalence between the two definitions:

Proposition 5.4. $\mathsf{RCA}_0 \vdash \forall$ *DBA* $\mathcal{A}.((5.5) \equiv (5.6))$.

Proof sketch. Let $\mathcal{A} = (A, Q, \delta, q_0, F)$ be a deterministic automaton. For the left-right implication let $Y \in Q^\omega$ be an 'accepting run' of X on $\mathcal{A}$ and use induction to show that $Y(x, q_X(x))$. For the right-left implication, we use comprehension to define an 'accepting run' $Y \in Q^\omega$ by: $Y(x, q) \ \equiv \ q_X(x, q)$. Clearly the definition of Y is Δ^0_1, and we can show that such Y is a 'correct run' by induction on x. □

Notice that, for a deterministic automaton, the formula for acceptance is *arithmetical* in X, i.e. there are no SO quantifiers. This will be rather important for uniformity in the simulation of cyclic proofs in the next section.

5.2. **Formalisations of some automaton constructions.** Recall that we may freely add symbols for primitive recursive functions to our language. Since we rely on various results from [KMPS19] as the 'engine' behind some of our proofs, we will use their notions for manipulating automata.

For NBA $\mathcal{A}, \mathcal{A}'$, we write $\mathcal{A}^c$ and $\mathcal{A} \sqcup \mathcal{A}'$ to denote the complement and union constructions of automata from [KMPS19] (Sects. 5 and 6 resp.). We also write $\text{Empty}(\mathcal{A})$ for the recursive algorithm from [KMPS19] (Sect. 6), expressed as a Σ_1 formula in $\mathcal{A}$, determining whether $\mathcal{A}$ computes the empty language. It will also be useful for us later, in order to bound logical and proof complexity, to notice that DBA can already be complemented in RCA_0. This is a rather unsurprising result but does not appear in [KMPS19], so we give it here.

For a DBA $\mathcal{A} = (A, Q, \delta, q_0, F)$, we define a complementary NBA $\mathcal{A}^c$ as follows,

$$\mathcal{A}^c \;:=\; (A, (Q \times \{0\}) \cup ((Q \setminus F) \times \{1\}), \delta^c, (q_0, 0), (Q \setminus F) \times \{1\})$$

where $\delta^c \subseteq (Q^c \times A) \times Q^c$ (writing Q^c for $Q \times \{0\} \cup (Q \setminus F) \times \{1\}$) is defined as:

$$\begin{array}{rl} & \{((q,0), a, (q',0)) \;:\; (q, a, q') \in \delta\} \\ \cup & \{((q,i), a, (q',1)) \;:\; (q, a, q') \in \delta, i = 0, 1, q' \in Q \setminus F\} \end{array}$$

The idea behind this construction is that a run of $\mathcal{A}^c$ follows $\mathcal{A}$ freely for some finite time (in the '0' component), after which it may no longer visit final states of $\mathcal{A}$ (once in the '1' component). The determinism of $\mathcal{A}$ guarantees that such a word is not accepted by it.

By directly inspecting the definitions from [KMPS19], and DBA complementation above, we have the following properties:

Observation 5.5. For NBA $\mathcal{A}, \mathcal{A}'$ we have that:

(1) $\text{Empty}(\mathcal{A})$ is a polynomial-time predicate in $\mathcal{A}$.
(2) $\mathcal{A} \sqcup \mathcal{A}'$ is constructible in polynomial-time from $\mathcal{A}$ and $\mathcal{A}'$.
(3) $\mathcal{A}^c$ is constructible in exponential-time from $\mathcal{A}$.

For a DBA $\mathcal{A}$, we have that:

(4) $\mathcal{A}^c$ is constructible in polynomial-time from $\mathcal{A}$.

None of these bounds are surprising, due to known bounds on the complexity of union, complementation and emptiness checking for (non)deterministic Büchi automata. Nonetheless it is important to state them for the particular constructions used in this work for bounds on proof complexity later.

Lemma 5.6. *From* [KMPS19] *we have the following:*

(1) $\mathsf{RCA}_0 \vdash \forall$ *NBA* $\mathcal{A}.(\text{Empty}(\mathcal{A}) \equiv \forall X \in A^\omega . X \notin \mathcal{L}(\mathcal{A}))$.
(2) $\mathsf{RCA}_0 \vdash \forall$ *NBA* $\mathcal{A}_1, \mathcal{A}_2.(X \in \mathcal{L}(\mathcal{A}_1 \sqcup \mathcal{A}_2) \equiv (X \in \mathcal{L}(\mathcal{A}_1) \vee X \in \mathcal{L}(\mathcal{A}_2)))$.
(3) $\mathsf{RCA}_0 + \Sigma^0_2\text{-}\mathsf{IND} \vdash \forall$ *NBA* $\mathcal{A}.(X \in A^\omega \supset (X \in \mathcal{L}(\mathcal{A}^c) \equiv X \notin \mathcal{L}(\mathcal{A})))$.

We also have that:

(4) $\mathsf{RCA}_0 \vdash \forall$ *DBA* $\mathcal{A}$. $(X \in A^\omega \supset (X \in \mathcal{L}(\mathcal{A}^c) \equiv X \notin \mathcal{L}(\mathcal{A})))$.

Proof. 1, 2 and 3 follow from [KMPS19], namely from Prop. 6.1 and Lemma 5.2, so we give a proof of 4.

Working in RCA_0, let $\mathcal{A} = (A, Q, \delta, q_0, F)$ be a DBA. For the right-left implication, if $X \notin \mathcal{L}(\mathcal{A})$ then $\exists x . \forall x' > x.\ q_X(x) \notin F$, so let x_0 witness this existential. Now, define by comprehension the run $Y \in (Q^c)^\omega$ as follows:

$$Y(x, y) \;\equiv\; ((x \leq x_0 \wedge y = (q_X(x), 0)) \vee (x > x_0 \wedge y = (q_X(x), 1)))$$

Now, indeed $Y(0,(q_0,0))$, since $q_X(0) = q_0$, and Y is a correct run of X on $\mathcal{A}^c$ by considering separately the cases $x < x_0$, $x = x_0$ and $x > x_0$. Finally, for any x, Y hits a final state at $\max(x, x_0) + 1 > x$.

For the left-right implication, suppose $X \in \mathcal{L}(\mathcal{A}^c)$ and let $Y \in (Q^c)^\omega$ be an accepting run. By induction we have $\forall x.(Y(x) = (q_X(x), 0) \vee Y(x) = (q_X(x), 1))$. Now, Y must eventually hit an accepting state of $\mathcal{A}^c$, i.e. in the 1-component, say at position x_0. Again by induction, we may show that Y remains in the 1-component of $\mathcal{A}^c$ after x_0, and hence $q_X(x) \notin F$ for $x \geq x_0$, as required. □

6. An exponential simulation of CA in PA

In this section we will adapt Simpson's approach in [Sim17] for showing that $\mathsf{CA} \subseteq \mathsf{PA}$ into a *uniform* result in PA. This essentially constitutes a formalisation of the soundness argument, Prop. 3.4, in a SO theory conservative over the target fragment of PA. The 'uniformity' we aim for ensures that the possibly non-elementary blowup translating from SO proofs to FO proofs occurs once and for all for a single arithmetical theorem. Only then do we instantiate the theorem (inside PA) by the cyclic proof in question, leading to an only elementary blowup.

To give an idea of how the result is obtained, and how our exposition refines that of [Sim17], we take advantage of the following aspects of the soundness argument for cyclic proofs:

(a) The Büchi automaton accepting all infinite branches of a cyclic proof is, in fact, deterministic, and so we can express acceptance of an ω-word in this automaton arithmetically.
(b) A branch of invalid sequents and corresponding assignments, as in the proof of Prop. 3.4, can be uniformly generated from an initial unsatisfying assignment by an arithmetical formula.
(c) Since all inductions are only up to ω, we need only *arbitrarily often* progressing traces, rather than explicit infinitely progressing traces.

Together, these properties give us just enough 'wiggle room' to carry out the soundness argument in a sufficiently uniform way.

Throughout this section we will also carefully track how much quantifier complexity is used in theorem statements, since we will later modify this argument to obtain a converse result to Thm. 4.1.

6.1. An arithmetically uniform treatment of automata.

Referring to (c) above, we define an arithmetical corollary of NBA acceptance that is nonetheless sufficiently strong to formalise the soundness argument for cyclic proofs:

Definition 6.1 (Arithmetic acceptance)**.** Let $\mathcal{A} = (A, Q, \delta, q_0, F)$ be a NBA and $X \in A^\omega$, and temporarily write:

- $F(x) :=$ "x is a finite run of X on $\mathcal{A}$ ending at a final state".
- $E(z, x, y) :=$ "z extends x to a finite run of X on $\mathcal{A}$ hitting $\geq y$ final states"

We define:

$$\mathrm{ArAcc}(X, \mathcal{A}) \;:=\; X \in A^\omega \wedge \exists x.\,(F(x) \wedge \forall y.\exists z.E(z, x, y)) \tag{6.1}$$

For intuition, we may consider ω-regular expressions rather than automata, which are of the form $\sum_{i<n} e_i \cdot f_i^\omega$, for some $n \in \mathbb{N}$, without loss of generality. The formula ArAcc for this expression essentially recognises infinite words that have prefixes of the form $\sigma\tau_k$ for some $\sigma \in \mathcal{L}(e_i)$, for some $i < n$, and $\tau_k \in \mathcal{L}(f_i^k)$ for each $k \in \mathbb{N}$. Clearly the condition ArAcc is a (provable) consequence of acceptance itself:

Proposition 6.2. $\mathsf{RCA}_0 \vdash \forall \mathcal{A}.(X \in \mathcal{L}(\mathcal{A}) \supset \mathrm{ArAcc}(X, \mathcal{A}))$.

Proof. Working in RCA_0, fix $\mathcal{A} = (A, Q, \delta, q_0, F)$ and suppose $X \in \mathcal{L}(\mathcal{A})$. Let $Y \in Q^\omega$ be an 'accepting run' of X on $\mathcal{A}$, cf. (5.5). We may show that,

$$\exists z \in Q^*.\text{“}z \text{ is a finite prefix of } Y \text{ hitting} \geq y \text{ final states in } \mathcal{A}\text{”} \tag{6.2}$$

by Σ^0_1-induction on y, appealing to the unboundedness of final states in Y for both the base case and the inductive steps. Now, in the definition of ArAcc in (6.1), we set x to be the least such z for which $(6.2)[1/y]$ holds (again by induction), so that $F(x)$ from (6.1) holds. Thus, for any $y \in \mathbb{N}$, we may find an appropriate z making $E(z, x, y)$ in (6.1) true by appealing to (6.2). The fact that z extends x follows from leastness of x and that Y is a sequence, cf. (5.3) and (5.4). □

Let us write $\mathcal{A}_1 \sqsubseteq \mathcal{A}_2$ for $\mathrm{Empty}((\mathcal{A}_1^c \sqcup \mathcal{A}_2)^c)$. We may now present our main 'uniform' result needed to carry out our soundness proof in FO theories.

Theorem 6.3. $\mathsf{RCA}_0 + \Sigma^0_2\text{-}\mathsf{IND}$ *proves:*

$$\forall\ \mathit{DBA}\ \mathcal{A}_1, \forall\ \mathit{NBA}\ \mathcal{A}_2.\ ((\mathcal{A}_1 \sqsubseteq \mathcal{A}_2 \wedge X \in \mathcal{L}(\mathcal{A}_1)) \supset \mathrm{ArAcc}(X, \mathcal{A}_2)) \tag{6.3}$$

Proof. Working in $\mathsf{RCA}_0 + \Sigma^0_2\text{-}\mathsf{IND}$, let $\mathcal{A}_1$ be a DBA and $\mathcal{A}_2$ be a NBA such that $X \in \mathcal{L}(\mathcal{A}_1)$ and $\mathcal{A}_1 \sqsubseteq \mathcal{A}_2$. We have:

	$\mathrm{Empty}((\mathcal{A}_1^c \sqcup \mathcal{A}_2)^c)$	since $\mathcal{A}_1 \sqsubseteq \mathcal{A}_2$
$\Longrightarrow$	$\forall Y \in A^\omega.\ Y \notin \mathcal{L}((\mathcal{A}_1^c \sqcup \mathcal{A}_2)^c)$	by Lemma 5.6.1
$\Longrightarrow$	$\forall Y \in A^\omega.\ Y \in \mathcal{L}(\mathcal{A}_1^c \sqcup \mathcal{A}_2)$	by Lemma 5.6.3
$\Longrightarrow$	$\forall Y \in A^\omega.(Y \in \mathcal{L}(A_1^c) \vee Y \in \mathcal{L}(\mathcal{A}_2))$	by Lemma 5.6.2
$\Longrightarrow$	$\forall Y \in A^\omega.(Y \in \mathcal{L}(\mathcal{A}_1) \supset Y \in \mathcal{L}(\mathcal{A}_2))$	by Lemma 5.6.4
$\Longrightarrow$	$X \in \mathcal{L}(\mathcal{A}_2)$	since $X \in \mathcal{L}(\mathcal{A}_1)$
$\Longrightarrow$	$\mathrm{ArAcc}(X, \mathcal{A}_2)$	by Prop. 6.2. □

Noticing that DBA acceptance is also purely arithmetical in X (cf. (a)), by the conservativity result Prop. 5.2, we have:

Corollary 6.4. $\mathrm{I}\Sigma_2(X)$ *proves* (6.3).

6.2. **Formalising the soundness argument for cyclic proofs.** At this point we are able to mostly mimic the formalisation of the soundness argument from [Sim17], although we must further show that a branch of invalid sequents, cf. the proof of Prop. 3.4, is uniformly describable (cf. (b)).

For $n \geq 0$, let $\mathbb{N}, \rho \vDash_n \varphi$ be an appropriate Δ_{n+1} formula (provably in $\mathrm{I}\Sigma_{n+1}$) asserting that a formula φ is true in $\mathbb{N}$ under the assignment ρ of its free variables to natural numbers, as long as φ is a Boolean combination of Σ_n (or Π_n) formulae.[6] Formally, the formula $\mathbb{N}, \rho \vDash_n \varphi$

[6] If φ is not a Boolean combination of Σ_n formulae then $\mathbb{N}, \rho \vDash_n \varphi$ crashes and returns $\bot$.

takes as arguments the *codes* of ρ and φ, i.e. their Gödel numbers; the construction of such a formula for $\vDash_n$ is standard (see, e.g., [Bus98, Kay91, HP93]) and it has size polynomial in n. Importantly, there are $\mathrm{I}\Sigma_{n+1}$ proofs that $\vDash_n$ satisfies 'Tarski's truth conditions'. Writing $\mathrm{Bool}(\Phi)$ for the class of Boolean combinations of Φ-formulae, we have:

Proposition 6.5 (Properties of $\vDash_n$, see e.g. [HP93])**.** *For $n \geq 0$, the following Π_{n+1} formulae have $\mathrm{I}\Sigma_{n+1}$ proofs of size polynomial in n:*

(1) $\forall \varphi \in \mathrm{Bool}(\Sigma_n).\forall \rho.\ (\mathbb{N}, \rho \vDash_n \neg\varphi \ \equiv\ \mathbb{N}, \rho \nvDash_n \varphi)$.
(2) $\forall \varphi, \psi \in \mathrm{Bool}(\Sigma_n).\forall \rho.\ (\mathbb{N}, \rho \vDash_n (\varphi \vee \psi) \ \equiv\ (\mathbb{N}, \rho \vDash_n \varphi \ \vee\ \mathbb{N}, \rho \vDash \psi))$.
(3) $\forall \varphi, \psi \in \mathrm{Bool}(\Sigma_n).\forall \rho.\ (\mathbb{N}, \rho \vDash_n (\varphi \wedge \psi) \ \equiv\ (\mathbb{N}, \rho \vDash_n \varphi \ \wedge\ \mathbb{N}, \rho \vDash \psi))$.
(4) $\forall \varphi \in \Sigma_n.\forall \rho.\ (\mathbb{N}, \rho \vDash_n \exists x.\varphi \ \equiv\ \exists y.(\mathbb{N}, \rho \cup \{x \mapsto y\} \vDash_n \varphi))$.
(5) $\forall \varphi \in \Pi_n.\forall \rho.\ (\mathbb{N}, \rho \vDash_n \forall x.\varphi \ \equiv\ \forall y.(\mathbb{N}, \rho \cup \{x \mapsto y\} \vDash_n \varphi))$.

We also have $\mathrm{I}\Sigma_{n+1}$ proofs of size polynomial in n of the substitution property*:*

(6) $\forall \varphi \in \mathrm{Bool}(\Sigma_n).\forall \rho.\forall$ *terms* t. $(\mathbb{N}, \rho \cup \{a \mapsto \rho(t)\} \vDash_n \varphi \ \equiv\ \mathbb{N}, \rho \vDash_n \varphi[t/a])$.

In particular we have the *reflection property*:

Proposition 6.6 (Reflection)**.** *For $n \geq 0$ we have $\mathrm{I}\Sigma_1 \vdash \varphi \equiv (\mathbb{N}, \varnothing \vDash_n \varphi)$ with proofs of size polynomial in n and $|\varphi|$, for any closed formula $\varphi \in \Sigma_n \cup \Pi_n$.*

Henceforth, all our proof complexity bounds in n follow from the fact that proofs are parametrised by $\vDash_n$ and its basic properties from Prop. 6.5 above.

Definition 6.7 (Uniform description of an invalid branch)**.** Let π be a CA preproof of a sequent $\Gamma \Rightarrow \Delta$, and let $n \in \mathbb{N}$ be such that all formulae occurring in π are Σ_n. Let ρ_0 be an assignment such that $\mathbb{N}, \rho_0 \vDash_n \bigwedge \Gamma$ but $\mathbb{N}, \rho_0 \nvDash_n \bigvee \Delta$. The branch of π **generated** by ρ_0 is the invalid branch as constructed in the proof of Prop. 3.4, where at each step that there is a choice of premiss the leftmost one is chosen, and at each step when there is a choice of assignment of a natural number to a free variable the least one is chosen. We write $\mathrm{Branch}_n(\pi, \rho_0, x, y)$ for the following predicate:

"the x^{th} element of the branch generated by ρ_0 in π is y"

To be precise, the 'element' y is given as a pair $\langle \rho_x, \Gamma_x \Rightarrow \Delta_x \rangle$ consisting of a sequent $\Gamma_x \Rightarrow \Delta_x$ and an assignment ρ_x that invalidates it.

Notice that $\mathrm{Branch}_n(\pi, \rho_0, x, y)$ is recursive w.r.t. the oracle $\vDash_n$, and so is expressible by a $\Delta_1(\vDash_n)$ formula, making it altogether Δ_{n+1} in its arguments. In fact, this is demonstrably the case in $\mathrm{I}\Sigma_{n+1}$, which can prove that $\mathrm{Branch}_n(\pi, \rho_0, -, -)$ is the graph of a *function*, as shown in Prop. 6.8 below.

Let us write $\mathsf{conc}(\pi)$ for the conclusion of a CA proof π and, as in Sect. 3.2, $\mathcal{A}_b^\pi$ and $\mathcal{A}_t^\pi$ for its branch and trace automata, resp. When we write $\mathbb{N}, \rho \vDash_n (\Gamma \Rightarrow \Delta)$ we mean the Δ_{n+1} formula $(\mathbb{N}, \rho \nvDash_n \bigwedge \Gamma) \vee (\mathbb{N}, \rho \vDash_n \bigvee \Delta)$.

Proposition 6.8. *For $n \geq 0$, there are $\mathrm{I}\Sigma_{n+1}$ proofs of size polynomial in n of:*

$$\begin{array}{l} \forall \pi \text{ a } \mathsf{CA} \text{ preproof containing only } \Sigma_n \text{ formulae.} \\ \forall \rho_0.\ ((\mathbb{N}, \rho_0 \nvDash_n \mathsf{conc}(\pi)) \supset \mathrm{Branch}_n(\pi, \rho_0, -, -) \in \mathcal{L}(\mathcal{A}_b^\pi)) \end{array} \tag{6.4}$$

Proof. Working in $\mathrm{I}\Sigma_{n+1}$, let π and ρ_0 satisfy the hypotheses of (6.4) above. The fact that $\mathrm{Branch}_n(\pi, \rho_0, -, -)$ is deterministic, cf. (5.4), follows directly by induction on the position

of the branch. The difficult part is to show that $\mathrm{Branch}_n(\pi, \rho_0, -, -)$ is total, cf. (5.3), i.e. that it never reaches a deadlock. For this we show,

$$\forall x. \exists \langle \rho_x, \Gamma_x \Rightarrow \Delta_x \rangle.\ (\mathrm{Branch}_n(\pi, \rho_0, x, \langle \rho_x, \Gamma_x \Rightarrow \Delta_x \rangle) \wedge\ \mathbb{N}, \rho_x \not\models_n (\Gamma_x \Rightarrow \Delta_x)) \tag{6.5}$$

by Σ_{n+1}-induction on x. The base case, when $x = 0$, follows by assumption, so we proceed with the inductive case. For a given x let $\langle \rho_x, \Gamma_x \Rightarrow \Delta_x \rangle$ witness (6.5) above and let r be the rule instance in π that $\Gamma_x \Rightarrow \Delta_x$ concludes.

If r is a $\exists$-r step with associated term t, then there is only one premiss which we show remains false in the current assignment. This follows from:

$$\begin{array}{rcll}
\mathbb{N}, \rho \not\models_n \exists x. \varphi & \Longrightarrow & \mathbb{N}, \rho \models_n \forall x. \neg\varphi & \text{by Prop. 6.5.1} \\
& \Longrightarrow & \forall y. (\mathbb{N}, \rho \cup \{x \mapsto y\} \models_n \neg\varphi) & \text{by Prop. 6.5.5} \\
& \Longrightarrow & \mathbb{N}, \rho \cup \{x \mapsto \rho(t)\} \models_n \neg\varphi & \text{by pure logic} \\
& \Longrightarrow & \mathbb{N}, \rho \models_n \neg\varphi[t/x] & \text{by Prop. 6.5.6} \\
& \Longrightarrow & \mathbb{N}, \rho \not\models_n \varphi[t/x] & \text{by Prop. 6.5.1.}
\end{array}$$

If r is a $\forall$-r step then there is only one premiss, for which we show that the appropriate invalidating assignment exists. This follows from,

$$\begin{array}{rcll}
\mathbb{N}, \rho \not\models_n \forall x. \varphi & \Longrightarrow & \mathbb{N}, \rho \models_n \exists x. \neg\varphi & \text{by Prop. 6.5.1} \\
& \Longrightarrow & \exists y. (\mathbb{N}, \rho \cup \{x \mapsto y\} \models_n \neg\varphi) & \text{by Prop. 6.5.4} \\
& \Longrightarrow & \exists \text{ least } y. (\mathbb{N}, \rho \cup \{x \mapsto y\} \models_n \neg\varphi) & \text{by } \Sigma_{n+1}\text{-}\mathsf{IND} \\
& \Longrightarrow & \exists \text{ least } y. (\mathbb{N}, \rho \cup \{x \mapsto y\} \not\models_n \varphi) & \text{by Prop. 6.5.1.}
\end{array}$$

where, in the penultimate implication, we rely on the fact that the appropriate 'minimisation' property is provable in Σ_{n+1}-IND (see, e.g., [Bus98]).

If r is a left quantifier step then it is treated similarly to the two right quantifier cases above by De Morgan duality. If r is a propositional step then the treatment is simple, following directly from Prop. 6.5. If r is an initial sequent we immediately hit a contradiction, since all of the axioms are provably true in all assignments. If r is a substitution step, then the existence of the appropriate assignment follows directly from the substitution property, Prop. 6.5.6.

Finally, we may show that every state of the run of $\mathrm{Branch}_n(\pi, \rho_0, -, -)$ on $\mathcal{A}^\pi_b$ is final, by Σ_{n+1}-induction, since it always correctly follows a branch of π. Thus we have that $\mathrm{Branch}_n(\pi, \rho_0, -, -) \in \mathcal{L}(\mathcal{A}^\pi_b)$. □

Now we can give a formalised proof of the soundness of cyclic proofs:

Theorem 6.9 (Soundness of cyclic proofs, formalised)**.** *For $n \geq 0$, there are $\mathrm{I}\Sigma_{n+2}$ proofs of size polynomial in n of:*

$$\begin{array}{l}
\forall \pi \text{ a } \mathsf{CA} \text{ preproof containing only } \Sigma_n \text{ formulae.} \\
(\mathcal{A}^\pi_b \sqsubseteq \mathcal{A}^\pi_t \ \supset\ \forall \rho_0.\ \mathbb{N}, \rho_0 \models_n \mathsf{conc}(\pi))
\end{array} \tag{6.6}$$

Proof. First, instantiating X in Cor. 6.4 with a Δ_{n+1} formula φ_n yields $O(|\varphi_n|)$-size $\mathrm{I}\Sigma_{n+2}$ proofs of $(6.3)[\varphi_n/X]$. Hence, setting φ_n to be $\mathrm{Branch}_n(\pi, \rho_0, -, -)$ and appealing to Prop. 6.8 above, we arrive at $\mathrm{I}\Sigma_{n+2}$ proofs of size polynomial in n of:

$$\begin{array}{l}
\forall \pi \text{ a } \mathsf{CA} \text{ preproof containing only } \Sigma_n \text{ formulae.} \\
\mathcal{A}^\pi_b \sqsubseteq \mathcal{A}^\pi_t \supset \forall \rho_0. (\mathbb{N}, \rho_0 \not\models_n \mathsf{conc}(\pi) \supset \mathrm{ArAcc}(\mathrm{Branch}_n(\pi, \rho_0, -, -), \mathcal{A}^\pi_t))
\end{array} \tag{6.7}$$

Now, working in $\mathrm{I}\Sigma_{n+2}$, to prove (6.6) let π satisfy $\mathcal{A}^\pi_b \sqsubseteq \mathcal{A}^\pi_t$. For contradiction assume, for some ρ_0, that $\mathbb{N}, \rho_0 \not\models_n \mathsf{conc}(\pi)$. By Prop. 6.8 we have $\mathrm{Branch}_n(\pi, \rho_0, -, -) \in \mathcal{L}(\mathcal{A}^\pi_b)$, so

we henceforth write $\Gamma_x \Rightarrow \Delta_x$ and ρ_x for the sequent and assignment at the x^{th} position of $\text{Branch}_n(\pi, \rho_0, -, -)$. By (6.7) above we have $\text{ArAcc}(\text{Branch}_n(\pi, \rho_0, -, -), \mathcal{A}^\pi_t)$, so let x witness its outer existential, cf. (6.1). Now, let y be the maximum value of $\rho_x(t)$ for all terms t occurring in $\Gamma_x \Rightarrow \Delta_x$. Again by $\text{ArAcc}(\text{Branch}_n(\pi, \rho_0, -, -), \mathcal{A}^\pi_t)$, we have that there is some (finite) trace z beginning from $\Gamma_x \Rightarrow \Delta_x$ that progresses $y + 1$ times. Writing $z(i)$ to denote the i^{th} term in the trace z, we may show by induction on $i \leq |z|$ that, if there are j progress points between $z(0)$ and $z(i)$, then we have that $\rho_x(z(0)) \geq \rho_{x+i}(z(i)) + j$. In particular, $y \geq \rho_x(z(0)) \geq \rho_{x+|z|}(z(|z|)) + (y + 1) \geq y + 1$, yielding a contradiction. □

6.3. PA exponentially simulates CA.

We can now give our main proof complexity result:

Theorem 6.10. *If π is a CA proof of φ, then we can construct a PA proof of φ of size exponential in $|\pi|$.*

Proof. Take the least $n \in \mathbb{N}$ such that π contains only Σ_n formulae; in particular $n \leq |\pi|$. Since π is a correct cyclic proof, there is a PA proof of $\mathcal{A}^\pi_b \sqsubseteq \mathcal{A}^\pi_t$, by exhaustive search. In fact, such a proof in Q may be constructed in exponential time in $|\pi|$, thanks to Obs. 5.5 and Σ_1-completeness of Q (see, e.g., [HP93]). Hence, by instantiating π in Thm. 6.9, we have $\text{I}\Sigma_{n+2}$ proofs of $\mathbb{N}, \varnothing \vDash_n \varphi$ of size exponential in $|\pi|$. Finally by the reflection property, Prop. 6.6, we have that $\text{I}\Sigma_{n+2} \vdash \varphi$ with proofs of size exponential in $|\pi|$. □

Notice that we already have a converse polynomial simulation of PA in CA by the results of [Sim17] or, alternatively, by Prop. 3.7.

7. $\text{I}\Sigma_{n+1}$ contains $\text{C}\Sigma_n$

In fact the proof method we developed in the last section allows us to recover a result on logical complexity too. By tracking precisely all the bounds therein, we obtain that $\text{C}\Sigma_n$ is contained in $\text{I}\Sigma_{n+2}$, which is already an improvement to Simpson's result (see Sect. 10 for a comparison). To derive such bounds, in this section we concern ourselves only with cyclic proofs containing Σ_n formulae. The universal closures of the conclusions of such proofs axiomatise $\text{C}\Sigma_n$, cf. Dfn. 3.5, so more complex theorems of $\text{C}\Sigma_n$ are thence derivable by pure logic.

In fact, we may actually improve this logical bound and arrive at an optimal result (given Thm. 4.1). By more carefully analysing the proof methods of [KMPS19], namely an inspection of the proofs of Thms. 5 and 12 in that work, we have that:

Proposition 7.1 (Implicit in [KMPS19])**.** $\text{RCA}_0 \vdash \forall X \in A^\omega.(X \in \mathcal{L}(\mathcal{A}^c) \equiv X \notin \mathcal{L}(\mathcal{A}))$*, for any NBA $\mathcal{A}$.*

Notice here that the universal quantification over NBA is *external*, so that the complementation proofs are not necessarily uniform. This is not a trivial result, since it relies on a version of Ramsey's theorem, the *additive Ramsey theorem*, which can be proved by induction on the number of 'colours'. Usual forms of Ramsey's theorem are not proved by such an argument, and in fact it is well known that RCA_0 cannot even prove Ramsey's theorem for pairs with only two colours (see, e.g., [Hir14]). We include in App. B a self-contained (and somewhat simpler) proof of Prop. 7.1 above, for completeness.

This allows us to 'un-uniformise' the results of the previous section, using Prop. 7.1 above instead of Lemma 5.6.3, in order to 'trade off' proof complexity for logical complexity:

Proposition 7.2 (Soundness of cyclic proofs, non-uniformly formalised)**.** *Let $n \geq 0$ and π be a* CA *proof containing only Σ_n formulae.* $\mathrm{I}\Sigma_{n+1} \vdash \forall \rho_0.(\mathbb{N}, \rho_0 \vDash_n \mathsf{conc}(\pi))$.

Proof sketch. We mimic the entire argument of Thm. 6.9 by instantiating the fixed proof π and using Prop. 7.1 above instead of Lemma 5.6.3. In particular, the required 'non-uniform' versions of Thm. 6.3 and Cor. 6.4 become derivable in RCA_0 and $\mathrm{I}\Sigma_1(X)$ resp., thus reducing the global induction complexity by one level. Hence we arrive at a 'non-uniform' version of Thm. 6.9, peculiar to the fixed proof π we began with, proved entirely within $\mathrm{I}\Sigma_{n+1}$, as required. □

Theorem 7.3. *For $n \geq 0$, we have that $\mathrm{C}\Sigma_n \subseteq \mathrm{I}\Sigma_{n+1}$.*

Proof. By the definition of $\mathrm{C}\Sigma_n$, cf. Dfn. 3.5, it suffices to derive in $\mathrm{I}\Sigma_{n+1}$ just the (possibly open) conclusions of $\mathrm{C}\Sigma_n$ proofs containing only Σ_n formulae, and so the result follows directly from Props. 7.2 and 6.6. □

7.1. **On the proof complexity of $\mathrm{C}\Sigma_n$.** One might be tempted to conclude that the elementary simulation of CA by PA should go through already for $\mathrm{C}\Sigma_n$ by $\mathrm{I}\Sigma_{n+2}$ (independently of n), due to the bounds implicit in the proof of Thm. 6.10. Furthermore, if we are willing to give up a few more exponentials in complexity, one may even bound the size of $\mathrm{I}\Sigma_1$ proofs arising from Prop. 7.1 by an appropriate elementary function (though this analysis is beyond the scope of this paper).

However, we must be conscious of the 'robustness' of the definition of $\mathrm{C}\Sigma_n$ proofs in terms of complexity. The one we gave, which essentially requires cyclic proofs to contain only Σ_n-formulae, is more similar to 'free-cut free' $\mathrm{I}\Sigma_n$ proofs than general ones, cf. Thm. 2.5, so it seems unfair to compare these notions of proof in terms of proof complexity. In fact we may define a more natural notion of a $\mathrm{C}\Sigma_n$ proof from the point of view of complexity, while inducing the same theory.

First, let us recall some notions from, e.g., [Bro06, BS11]. Any cyclic proof can be written in 'cycle normal form', where any 'backpointer' (e.g., the conclusions of upper sequents marked $\bullet$ until now) points only to a sequent that has occurred below it in the proof. Referring to the terminology of [Bro06, BS11] etc., we say that a sequent is a *bud* in a cyclic proof if it has a backpointer pointing to an identical sequent (called the *companion*).

Proposition 7.4. *If φ has a* CA *proof whose buds and companions contain only Σ_n formulae then $\mathrm{C}\Sigma_n \vdash \varphi$.*

Proof idea. Once again, we simply apply free-cut elimination, Thm. 2.5, treating any backpointers as 'initial sequents' for a proof in cycle normal form. □

This result again shows the robustness of the definition of $\mathrm{C}\Sigma_n$ as a theory, and we would further argue that, from the point of view of proof complexity, the backpointer-condition induced by Prop. 7.4 above constitutes a better notion of 'proof' for $\mathrm{C}\Sigma_n$. One could go yet further and argue that an even better notion of 'proof' would allow pointers to any other identical sequent in the proof (not just those below it), which could potentially make for an exponential improvement in proof complexity.

At the same time we see that it is not easy to compare the proof complexity of such systems for $\mathrm{C}\Sigma_n$ with those for $\mathrm{I}\Sigma_{n+1}$, due to the fact that we have used a free-cut elimination result for the simulations in both directions, inducing a possibly non-elementary blowup

in proof size. It would be interesting if a more fine-grained result regarding the relative proof complexity of $\mathrm{I}\Sigma_{n+1}$ and $\mathrm{C}\Sigma_n$ could be established, but this is beyond the scope of the current work.

8. Some metamathematical results

In this section we make some further observations on various properties of the cyclic theories in this work, which are later applied in Sect. 9. The exposition we give is brief, since we follow standard methods, but we provide appropriate references for the reader.

8.1. **Provably recursive functions of $\mathrm{C}\Delta_0$.** As a corollary of the Thms. 4.1 and 7.3 we have that, for $n \geq 1$, the provably recursive functions of $\mathrm{C}\Sigma_n$ coincide with those of $\mathrm{I}\Sigma_{n+1}$. Such functions were characterised by Parsons in [Par72] as certain fragments of Gödel's system T or, equivalently, by recursion up to suitably bounded towers of ω, e.g. in [Bus95]. This apparently leaves open a gap for the case of $\mathrm{C}\Delta_0$ (a.k.a. $\mathrm{C}\Sigma_0$). However notice that from the definition of $\mathrm{C}\Sigma_n$ and Thm. 4.1 we have:

Corollary 8.1 (of Thm. 4.1)**.** *$\mathrm{C}\Sigma_n$ is axiomatised by the Π_{n+1}-consequences of $\mathrm{I}\Sigma_{n+1}$, for $n \geq 0$.*

In particular, since $\mathrm{I}\Delta_0$ is known to be Π_1-axiomatised (see, e.g., Thm. 1.27 in [HP93]), we have that in fact $\mathrm{C}\Delta_0 \supseteq \mathrm{I}\Delta_0$ (i.e. over *all* theorems). Thus $\mathrm{C}\Delta_0$ can prove recursive *at least* the functions (bitwise computable) in the linear-time hierarchy, from the analogous result for $\mathrm{I}\Delta_0$ (see, e.g., Theorem III.4.8 in [CN10]). Conversely, since $\mathrm{C}\Delta_0$ is also Π_1-axiomatised by the above observation, it has no more 'computational content' than $\mathrm{I}\Delta_0$, as we will now see.

Recall that the linear-time hierarchy (**LTH**) is the class of predicates expressible by a Δ_0 formula in the language of arithmetic.[7] A function is said to be in **FLTH** just if it has linear growth rate (in terms of the bit string representation) and is bitwise computable in **LTH**.

Proposition 8.2. *If $\mathrm{C}\Delta_0 \vdash \forall \vec{x}.\exists! y.\varphi(\vec{x}, y)$ for a Σ_1-formula φ, then φ computes the graph of a function in* **FLTH**.

Proof sketch. Suppose φ is $\exists \vec{y}.\varphi_0$ with $\varphi_0 \in \Delta_0$. By 'Parikh's theorem' for Π_1-axiomatised theories (see, e.g., Thm. III.2.3 in [CN10]) we moreover have $\mathrm{C}\Delta_0 \vdash \forall \vec{x}.\exists! y < t.\exists \vec{y} < \vec{t}.\varphi_0$ for some terms $t, \vec{t}$. This means we may simply search for a witness $y < t$ of linear size verifying $\exists \vec{y} < \vec{t}.\varphi_0$, an **LTH** property, and thus $\exists \vec{y} < \vec{t}.\varphi_0$ computes the graph of a function in **FLTH**. □

Interestingly, we cannot strengthen the above proposition to "$\mathrm{C}\Delta_0$ and $\mathrm{I}\Delta_0$ prove the same Π_2 theorems". This is because $\mathrm{I}\Sigma_1$ proves the *consistency* of $\mathrm{I}\Delta_0$, a Π_1 sentence, so $\mathrm{C}\Delta_0$ does too by Thm. 4.1. Thus the aforementioned stronger statement would contradict Gödel's second incompleteness theorem for $\mathrm{I}\Delta_0$.

See, e.g., [Bus98, CN10] for further discussions on the provably recursive functions of fragments of (bounded) arithmetic, and see, e.g., [CK02] for more details on relationships between the language of arithmetic and recursive function classes.

[7] Equivalently it is the class of predicates recognised by an alternating Turing machine with random access in a linear number of steps with only boundedly many alternations. See, e.g., [CK02] for more details.

8.2. **Failure of cut-admissibility.** As a corollary of our results, we may formally conclude that the cut rule is not admissible in CA, or indeed any of its fragments $\mathrm{C}\Sigma_n$.[8] In fact the situation is rather worse than that:

Corollary 8.3 (of Thms. 4.1 and 7.3)**.** *Let $n \geq 1$. The class of CA proofs with only Σ_{n-1} cuts is not complete for even the Π_1 theorems of $\mathrm{C}\Sigma_n$.*

Proof. For a recursively axiomatised theory T, let $\mathsf{Con}(T)$ be an appropriate Π_1 sentence expressing that "T does not prove $0 = 1$". It is well-known that $\mathrm{I}\Sigma_{n+1} \vdash \mathsf{Con}(\mathrm{I}\Sigma_n)$ (see, e.g., [Kay91, Bus98, HP93]), so also $\mathrm{C}\Sigma_n \vdash \mathsf{Con}(\mathrm{I}\Sigma_n)$ by Thm. 4.1. For contradiction, suppose $\mathsf{Con}(\mathrm{I}\Sigma_n)$ concludes some CA proof with only Σ_{n-1} cuts; then in fact $\mathrm{C}\Sigma_{n-1} \vdash \mathsf{Con}(\mathrm{I}\Sigma_n)$ by degeneralising (for the case $n = 1$) and the subformula property. However this implies $\mathrm{I}\Sigma_n \vdash \mathsf{Con}(\mathrm{I}\Sigma_n)$ by Thm. 7.3, which is impossible by Gödel's second incompleteness theorem for $\mathrm{I}\Sigma_n$. □

See, e.g., [Bus98, Kay91, HP93] for further discussions on the provability of consistency principles for fragments of arithmetic.

8.3. **Reflection and consistency.** Thanks to the uniformity of the results from Sect. 6, we can give some fundamental metalogical properties regarding provable soundness and consistency of cyclic proofs. First we will fix our formalisation of $\mathrm{C}\Sigma_n$-provability (of arbitrary sequents, not just Σ_n) in the language of arithmetic.

Let $n \geq 0$. We will fix some appropriate formula $\mathsf{Prf}_n(\pi, \varphi)$ expressing that π is a $\mathrm{C}\Sigma_n$ proof of φ. We suppose that proofs are written as usual derivations (finite trees or dags) whose leaves are either axiom instances from Q, or otherwise some (possibly open) Σ_n sequent labelled by an associated cyclic proof that derives it (containing only Σ_n formulae). Descriptively, $\mathsf{Prf}_n(x, y)$ checks that π is a proof of φ by first checking that it is a well-formed derivation, then checking that each premiss is either an axiom instance from Q or is labelled by a correct cyclic proof deriving it. In the latter case it must search for a certificate verifying that the cyclic proof satisfies the automaton-inclusion condition, i.e. that $\mathcal{A}_b \sqsubseteq \mathcal{A}_t$.

While $\mathsf{Prf}_n(\pi, \varphi)$ is recursive in π and φ, this may not be provably the case in weak theories such as $\mathrm{I}\Delta_0$. Thus we fix Prf_n to be an appropriate Σ_1 formula, as described above, and we write $\Box_n\varphi$ for $\exists\pi.\mathsf{Prf}_n(\pi, \varphi)$. We write Π_k-$\mathsf{Rfn}(\mathrm{C}\Sigma_n)$ for the (local) Π_k-*reflection principle* of $\mathrm{C}\Sigma_n$. I.e.

$$\Pi_k\text{-}\mathsf{Rfn}(\mathrm{C}\Sigma_n) \;:=\; \{\Box_n\varphi \supset \varphi : \varphi \in \Pi_k\}$$

Corollary 8.4 (of Thm. 6.9)**.** *For $n \geq 0$, we have $\mathrm{I}\Sigma_{n+2} \vdash \Pi_{n+1}$-$\mathsf{Rfn}(\mathrm{C}\Sigma_n)$.*

Proof. Let $\varphi(\vec{x})$ be a Σ_n formula. Working in $\mathrm{I}\Sigma_{n+2}$, suppose that $\Box_n\forall\vec{x}.\varphi$ (so that $\mathrm{C}\Sigma_n \vdash \forall\vec{x}.\varphi(\vec{x})$). We may assume that every formula occurring in a $\mathrm{C}\Sigma_n$ proof of the sequent $\Rightarrow \varphi(\vec{x})$ is Σ_n, thanks to free-cut elimination, Thm. 2.5 (recall that this result is provable already in $\mathrm{I}\Sigma_1$). Thus we have $\mathbb{N}, \varnothing \vDash_{n+1} \forall\vec{x}.\varphi(\vec{x})$ by Thm. 6.9 and Prop. 6.5.5, whence the result follows by the reflection property, Prop. 6.6. □

Notice that, while the statement of Cor. 8.4 above is peculiar to the current formulation of a $\mathrm{C}\Sigma_n$ proof, Prf_n, it holds also under any other notion of proof that is provably equivalent in $\mathrm{I}\Sigma_{n+2}$. In particular, in Sect. 7.1 we discussed another notion of proof for $\mathrm{C}\Sigma_n$ which, morally, allowed "free cuts" to occur inside cycles. Since the equivalence of the two formulations,

[8] This observation was pointed out to me by Stefano Berardi.

Prop. 7.4, is proved using only free-cut elimination and basic reasoning, all formalisable in $\mathrm{I}\Sigma_1$, the version of Cor. 8.4 for that more liberal notion of a $\mathrm{C}\Sigma_n$ proof holds too.

As usual, we may see Π_1-reflection as another formulation of 'consistency'. Let us write $\mathsf{Con}(\mathrm{C}\Sigma_n)$ for the sentence $\neg\Box_n 0 = 1$ (notice that this is a Π_1 sentence).

Corollary 8.5 (of Thm. 6.9)**.** *For $n \geq 0$, we have $\mathrm{I}\Sigma_{n+2} \vdash \mathsf{Con}(\mathrm{C}\Sigma_n)$*

Proof. Follows immediately from Cor. 8.4 above by substituting $0 = 1$ for φ. □

We will see in the next subsection that this result is, in fact, optimal with respect to logical complexity.

8.4. **Incompleteness.** Unsurprisingly, all the theories $\mathrm{C}\Sigma_n$ suffer from Gödel's incompleteness theorems. Even though $\mathrm{C}\Sigma_n$ is not explicitly defined axiomatically, it does have a recursively enumerable notion of provability, namely $\Box_n$, and so must be incomplete with respect to this notion (see, e.g., Thm. 2.21 in [HP93]):

Theorem 8.6 (Gödel's second incompleteness theorem, for cyclic theories)**.** *For $n \geq 0$, as long as $\mathrm{C}\Sigma_n$ is consistent (i.e. $\mathrm{C}\Sigma_n \nvdash 0 = 1$), we have $\mathrm{C}\Sigma_n \nvdash \mathsf{Con}(\mathrm{C}\Sigma_n)$.*

Consequently we have that Cor. 8.5 is, in fact, optimal in terms of logical complexity:

Corollary 8.7. *For $n \geq 0$, we have $\mathrm{I}\Sigma_{n+1} \nvdash \mathsf{Con}(\mathrm{C}\Sigma_n)$.*

Proof. Suppose otherwise. Then also $\mathrm{C}\Sigma_n \vdash \mathsf{Con}(\mathrm{C}\Sigma_n)$ by Π_1-conservativity, cf. Thm. 4.1, which contradicts Gödel's second incompleteness above, Thm. 8.6. □

We will see in the next section that this has a curious consequence for the reverse mathematics of results in ω-automaton theory.

9. On the logical strength of McNaughton's theorem

In this section we show how the results of this work yield an unexpected corollary: certain formulations of *McNaugton's theorem*, that every NBA has an equivalent deterministic 'parity' or 'Muller' automaton, are not provable in RCA_0. The general question of the logical strength of McNaughton's theorem was notably left open in the recent work [KMPS19].

Our result is non-uniform in the sense that unprovability holds for any *explicit* primitive recursive determinisation construction. As far as the author is aware this accounts for all known proofs of McNaughton's theorem, suggesting that it is unlikely to be provable at all, in its usual uniform version, in RCA_0. That said, we point out that the statement of McNaughton's theorem itself is arguably not so well-defined in the context of reverse mathematics: it is not clear in RCA_0 that different versions of the theorem coincide, namely with respect to the choice of (a) acceptance conditions (parity, Muller, etc.) and (b) formulation of the set of states infinitely often hit during a run (negative, $\forall$, vs. positive, $\exists$).

Our argument is based on an alternative route to proving the soundness of $\mathrm{C}\Sigma_n$. Assuming that an appropriate version of McNaughton's theorem is indeed provable in RCA_0, we are in fact able to formalise the soundness argument for $\mathrm{C}\Sigma_n$ already in $\mathrm{I}\Sigma_{n+1}$. However, consequently we have that $\mathrm{I}\Sigma_{n+1}$ proves the consistency of $\mathrm{C}\Sigma_n$, and so $\mathrm{C}\Sigma_n$ proves its own consistency by Π_1-conservativity, cf. Thm. 4.1, which is absurd by Gödel's second incompleteness theorem for $\mathrm{C}\Sigma_n$, Thm. 8.6.

9.1. **Deterministic parity automata and universality.** Due to space considerations, we only briefly present the details of parity automata. The reader is encouraged to consult, e.g., [Tho97], for further details on automaton theory for ω-languages.

A (non-deterministic) **Rabin** or **parity** automaton (NRA) is a just a NBA where, instead of a set of final states F, we have a function $c : Q \to \mathbb{N}$, called a **colouring**. A word is accepted by a NRA if it has a run in which the least colour of a state occurring infinitely often is even. The notion of deterministic parity automaton (DRA) is analogous to that of a DBA, i.e. requiring the transition relation to be deterministic and total.

Theorem 9.1 McNaughton, [McN66]**.** *For every NBA $\mathcal{A}$, we can effectively construct a DRA accepting the same language.*

Actually, McNaughton gave this result for deterministic *Muller* automata rather than parity automata. The equivalence of these two models is well-known though, as we previously mentioned, it is not clear whether RCA_0 can prove their equivalence. The fact that we use parity automata here is arbitrary; we believe a similar exposition could be carried out for Muller automata.

As for DBA, we may naturally express language acceptance for a DRA $\mathcal{A} = (A, Q, \delta, q_0, c)$ by an arithmetical formula, i.e. without SO quantifiers. For our purposes, it will be useful to take a 'negative' formulation of acceptance:

$$X \in \mathcal{L}(\mathcal{A}) \;:=\; \forall q \in Q. \left(\left(\begin{array}{ll} & \forall x. \exists x' > x.\ q_X(x') = q \\ \wedge & \exists x. \forall x' > x.\ c(q_X(x')) \geq c(q) \end{array} \right) \supset \text{“}c(q) \text{ is even”} \right)$$

We write $\sigma : q_1 \xrightarrow[\delta]{*} q_2$ if a word $\sigma \in A^*$ determines a path along δ starting at q_1 and ending at q_2. We write $\xrightarrow[\delta]{+}$ when the path is nonempty. A *simple loop* about a state $q \in Q$ is a nonempty path along δ beginning and ending at q that visits no intermediate state more than once.

Recall that we call an ω-automaton *universal* if it accepts all ω-words over its alphabet. We write $\mathrm{Univ}(\mathcal{A})$ for a standard recursive procedure testing universality of a DRA $\mathcal{A}$: "for every odd-coloured state q reachable from q_0, any simple loop about q contains a state coloured by an even number $< c(q)$". More formally, writing $\sigma' \leq \sigma$ if σ' is a prefix of σ:

$$\mathrm{Univ}(\mathcal{A}) \;:=\; \begin{array}{l} \forall q \xleftarrow[\delta]{*} q_0.\ \forall \sigma : q \xrightarrow[\delta]{+} q. \\ \left(\begin{array}{ll} & \text{“}\sigma \text{ is a simple loop”} \wedge \text{“}c(q) \text{ is odd”} \\ \supset & \exists \sigma' \leq \sigma.\ \exists q' \in Q.\ (\sigma' : q \xrightarrow[\delta]{+} q' \ \wedge \text{“}c(q') \text{ even”} \wedge c(q') < c(q)) \end{array} \right) \end{array}$$

Clearly this formula is provably Δ^0_1 in RCA_0. Furthermore:

Proposition 9.2. $\mathsf{RCA}_0 \vdash \forall$ *DRA* $\mathcal{A}$. $(\mathrm{Univ}(\mathcal{A}) \equiv \forall X \in A^\omega . X \in \mathcal{L}(\mathcal{A}))$.

Proof. Working in RCA_0, let $\mathcal{A} = (A, Q, \delta, q_0, c)$ be a DRA. For the left-right implication, suppose there is some $X \in A^\omega$ such that $X \notin \mathcal{L}(\mathcal{A})$. Thus we have some $q \in Q$ such that $c(q)$ is odd and the following hold:

$$\forall x. \exists x' > x.\ q_X(x') = q \tag{9.1}$$

$$\exists x. \forall x' > x.\ c(q_X(x')) \geq c(q) \tag{9.2}$$

Let x_0 be a witness to (9.2), and let $x_0 < x_1 < x_2$ such that $q_X(x_1) = q_X(x_2) = q$, by two applications of (9.1). We will need the following intermediate (arithmetical) result,

$$\text{If } \sigma : q \xrightarrow[\delta]{+} q, \text{ there is a subsequence } \sigma' \text{ of } \sigma \text{ that is a simple loop on } q.$$

which follows directly by induction on $|\sigma|$, eliminating intermediate loops at each inductive step in the case of non-simplicity. Now we apply this result to the sequence $(X(x))_{x=x_1}^{x_2}$ to obtain a simple loop about q; moreover since this will be a subsequence of $(X(x))_{x=x_1}^{x_2}$, we have that any even-coloured state occurring in it is coloured $> c(q)$, since x_0 witnesses (9.2) and $x_0 < x_1 < x_2$, so $\neg\mathrm{Univ}(\mathcal{A})$.

For the right-left implication, we proceed again by contraposition. Suppose $\neg\mathrm{Univ}(\mathcal{A})$, and let $\sigma : q_0 \xrightarrow[\delta]{*} q$ and $\tau : q \xrightarrow[\delta]{+} q$ such that $c(q)$ is odd, and τ is a simple loop containing no states coloured $< c(q)$. We may now set $X = \sigma\tau^\omega$ (which is easily defined by comprehension) and show that $X \notin \mathcal{L}(\mathcal{A})$. For this it suffices to show (9.1) and (9.2) above. For the former, given x we set $x' = |\sigma| + m|\tau| > x$, for some sufficiently large m. For the latter, we set $x = |\sigma|$ as the witness to the outer existential, whence (9.2) follows by construction of τ. □

9.2. **Reducing soundness of $\mathrm{C}\Delta_0$ to a version of McNaughton's theorem.** Henceforth we may write $\mathcal{L}(\mathcal{A}) = \mathcal{L}(\mathcal{B})$ as shorthand for $\forall X.(X \in \mathcal{L}(\mathcal{A}) \equiv X \in \mathcal{L}(\mathcal{B}))$, where $\mathcal{A}$ and $\mathcal{B}$ may be any type of automaton thus far encountered with respect to their associated notions of membership. Based on our 'negative' formulation of DRA acceptance, we define for a (definable) function d:

$$\mathsf{McNaughton}_d^- \;:=\; \forall \text{ NBA } \mathcal{A}.(\text{“}d(\mathcal{A}) \text{ is a DRA”} \wedge \mathcal{L}(\mathcal{A}) = \mathcal{L}(d(\mathcal{A}))$$

Assuming this is provable in RCA_0 for some primitive recursive function d, we will reproduce a version of Thm. 6.3 in $\mathrm{I}\Sigma_{n+1}$. The idea is that, rather than expressing the fact that $\mathcal{L}(\mathcal{A}_1) \subseteq \mathcal{L}(\mathcal{A}_2)$ by saying "$(\mathcal{A}_1^c \sqcup \mathcal{A}_2)^c$ is empty", as we did in Sects. 6 and 7, we may rather express it as "$\mathcal{A}_1^c \sqcup \mathcal{A}_2$ is universal", relying on $\mathsf{McNaughton}_d^-$ and Prop. 9.2 above.

For a DBA $\mathcal{A}_1$ and a NBA $\mathcal{A}_2$ we define $\mathcal{A}_1 \sqsubseteq_d \mathcal{A}_2$ as $\mathrm{Univ}(d(\mathcal{A}_1^c \sqcup \mathcal{A}_2)))$. Let us write (6.3_d) for the equation (6.3) with $\sqsubseteq$ replaced by $\sqsubseteq_d$, i.e.:

$$\forall \text{ DBA } \mathcal{A}_1, \forall \text{ NBA } \mathcal{A}_2.\ ((\mathcal{A}_1 \sqsubseteq_d \mathcal{A}_2 \wedge X \in \mathcal{L}(\mathcal{A}_1)) \supset \mathrm{ArAcc}(X, \mathcal{A}_2)) \tag{6.3_d}$$

We have the following analogue to Thm. 6.3:

Proposition 9.3. $\mathsf{RCA}_0 + \mathsf{McNaughton}_d^- \vdash (6.3_d)$.

Proof. Mimicking the proof of Thm. 6.3, we work in RCA_0 and suppose $X \in \mathcal{L}(\mathcal{A}_1)$ and $\mathcal{A}_1 \sqsubseteq_d \mathcal{A}_2$. We have:

	$\mathrm{Univ}(d(\mathcal{A}_1^c \sqcup \mathcal{A}_2))$	since $\mathcal{A}_1 \sqsubseteq_d \mathcal{A}_2$
$\Longrightarrow$	$\forall Y \in A^\omega.Y \in \mathcal{L}(\mathcal{A}_1^c \sqcup \mathcal{A}_2)$	by Prop. 9.2 and $\mathsf{McNaughton}_d^-$
$\Longrightarrow$	$\forall Y \in A^\omega.(Y \in \mathcal{L}(\mathcal{A}_1^c) \vee Y \in \mathcal{L}(\mathcal{A}_2))$	by Lemma 5.6.2
$\Longrightarrow$	$\forall Y \in A^\omega.(Y \in \mathcal{L}(\mathcal{A}_1) \supset Y \in \mathcal{L}(\mathcal{A}_2))$	by Lemma 5.6.4
$\Longrightarrow$	$X \in \mathcal{L}(\mathcal{A}_2)$	since $X \in \mathcal{L}(\mathcal{A}_1)$
$\Longrightarrow$	$\mathrm{ArAcc}(X, \mathcal{A}_2)$	by Prop. 6.2. □

We may use this result to reconstruct the entire formalised soundness argument for $\mathrm{C}\Sigma_n$ of Sect. 6 in $\mathrm{I}\Sigma_{n+1}$ instead of $\mathrm{I}\Sigma_{n+2}$, assuming $\mathsf{McNaughton}_d^-$. In particular, using Prop. 9.3 above instead of Thm. 6.3, we may recover versions of Cor. 6.4, Thm. 6.9 and Cor. 8.4 for $\mathrm{I}\Sigma_{n+1}$ instead of $\mathrm{I}\Sigma_{n+2}$, with respect to (6.3_d) instead of (6.3). Formally, let $\Pi_k\text{-}\mathsf{Rfn}_d(\mathrm{C}\Sigma_n)$ denote the formulation of the Π_k-reflection principle for $\mathrm{C}\Sigma_n$ induced by using $\sqsubseteq_d$ instead of $\sqsubseteq$ throughout Sect. 8.3, with respect to the definitions of Prf_n and $\Box_n$. Similarly, let $\mathsf{Con}_d(\mathrm{C}\Sigma_n)$ be the induced consistency principle.

Proposition 9.4. *For $n \geq 0$, if $\mathsf{RCA}_0 \vdash \mathsf{McNaughton}_d^-$ for some primitive recursive function d, then $\mathrm{I}\Sigma_{n+1} \vdash \Pi_{n+1}\text{-}\mathsf{Rfn}_d(\mathrm{C}\Sigma_n)$, so in particular $\mathrm{I}\Sigma_{n+1} \vdash \mathsf{Con}_d(\mathrm{C}\Sigma_n)$.*

Proof sketch. The argument goes through just like that of Cors. 8.4 and 8.5 of Thm. 6.9, except that we use Prop. 9.3 instead of Thm. 6.3. Appealing to the assumption that $\mathsf{RCA}_0 \vdash \mathsf{McNaughton}_d^-$, this version of the argument requires only $\Sigma^0_1\text{-}\mathsf{IND}$ instead of $\Sigma^0_2\text{-}\mathsf{IND}$, thus yielding $\mathrm{I}\Sigma_{n+1}$ proofs overall once we substitute the appropriate formulae for X. □

Theorem 9.5. *$\mathsf{RCA}_0 \nvdash \mathsf{McNaughton}_d^-$, for any primitive recursive function d.*

Proof sketch. The same argument as Cor. 8.7 holds for our revised notion of consistency; in particular we have that $\mathrm{I}\Sigma_1 \nvdash \mathsf{Con}_d(\mathrm{C}\Delta_0)$. The result now follows immediately by the contraposition of Prop. 9.4 above, for $n = 0$. □

10. Conclusions and further remarks

In this work we developed the theory of cyclic arithmetic by studying the logical complexity of its proofs. We showed that inductive and cyclic proofs of the same theorems require similar logical complexity, and obtained tight quantifier complexity bounds in both directions. We further showed that the proof complexity of the two frameworks differs only elementarily, although it remains unclear how to properly measure proof complexity for the fragments $\mathrm{C}\Sigma_n$, even if the theory itself seems well-defined and robust. Many of these issues constitute avenues for further work.

10.1. **Comparison to the proofs of** [BT17b] **and** [Sim17]**.** One reason for our improved quantifier complexity compared to [Sim17], is that Simpson rather relies on *Weak König's Lemma* (WKL) to obtain an infinite branch when formalising the soundness argument for cyclic proofs. This, *a priori*, increases quantifier complexity of the argument, since WKL is known to be unprovable in RCA_0 even in the presence of $\Sigma^0_2\text{-}\mathsf{IND}$; in fact, it is incomparable to $\Sigma^0_2\text{-}\mathsf{IND}$ (see, e.g., [KMPS19]). That said, we believe that the 'bounded-width WKL' (bwWKL) of [KMPS19] should suffice to carry out Simpson's proof, and this principle is provable already in $\mathsf{RCA}_0 + \Sigma^0_2\text{-}\mathsf{IND}$. Applying this strategy to his proof yields only that $\mathrm{C}\Sigma_n \subseteq \mathrm{I}\Sigma_{n+3}$, since bwWKL is applied to a Π^0_{n+1} set, though this should improve to a $\mathrm{I}\Sigma_{n+2}$ bound by using the non-uniform version of NBA complementation implicit in [KMPS19], cf. Prop. 7.1. We reiterate that the main improvement here is in giving a uniform formulation of those results; not only does this lead to a better proof complexity result, cf. Thm. 6.10, but we also recover a metamathematical account of the theories $\mathrm{C}\Sigma_n$, cf. 8.

Berardi and Tatsuta's approach, [BT17b], is rather interesting since it is arguably more 'structural' in nature, relying on proof-level manipulations rather than reflection principles. That said there are still crucial sources of logical complexity blowup, namely

in an 'arithmetical' version of *Ramsey's theorem* (Thm. 5.2) and the consequent *Podelski-Rybalchenko termination theorem* (Thm. 6.1). Both of these apparently increase quantifier complexity by several levels, and so their approach does not seem to yield comparable logical bounds to this work. Since proof complexity is not a primary consideration of their work, it is not simple to track the precise bounds in [BT17b]. There are some apparent sources of exponential blowups,[9] though it seems that the global simulation is elementary. As before, we reiterate that the major improvement in the present work is in the uniformity of our exposition: the approach of [BT17b] is fundamentally non-uniform so does not yield any metamathematical account of cyclic arithmetic.

10.2. **On the correctness criteria for cyclic proofs.** Since the algorithms used to check correctness of a cyclic preproof reduce to the inclusion of Büchi automata, the exponential simulation of CA by PA is optimal, unless there is a nondeterministic subexponential-time algorithm for **PSPACE** or, more interestingly, there is an easier way to check cyclic proof correctness. (In fact, technically, it would suffice to have an easier criterion for a *larger* class of preproofs that were, nonetheless, sound.) As far as we know, **PSPACE** remains the best known upper bound for checking the correctness of general cyclic preproofs, although efficient algorithms have recently been proposed for less general correctness criteria, cf. [Str17, NST18]. Thus, it would be interesting to prove a corresponding lower bound or otherwise improve the upper bound. Conditional such results could be obtained via, say, certain polynomial upper bounds on proof complexity in CA: for instance, if CA were to have polynomial-size proofs of each correct Büchi inclusion then cyclic proof correctness would not be polynomial-time checkable, unless **NP** = **PSPACE**. Unfortunately naïve attempts at this approach fail, but the general question of whether PA and CA are exponentially separated seems pertinent.

On the other hand, the translation of Lemma 4.2 from inductive proofs to cyclic proofs is rather structured. In light of the converse result in Sect. 6 it might make sense in further work, from the point of view of logical complexity, to consider only cyclic proofs accepted by some weaker more efficiently verified criterion, such as [Str17, NST18].

10.3. **Interpreting ordinary inductive definitions in arithmetic.** In earlier work by Brotherston and Simpson, cyclic proofs were rather considered over a system of FO logic extended by 'ordinary' *Martin-Löf* inductive definitions [ML71], known as FOL_{ID} [Bro06, BS07, BS11]. Berardi and Tatsuta showed in [BT17b] that the cyclic system CLKID^ω for FOL_{ID} is equivalent to the inductive system LKID, when at least arithmetic is present, somewhat generalising Simpson's result [Sim17]. We point out that ordinary Martin-Löf inductive definitions can be *interpreted* in arithmetic in the usual way by a Σ_1 inductive construction of 'approximants', and a proof of CLKID^ω may be similarly interpreted line-by-line in CA. (This is similar to the role of the 'stage number predicates' in [BT17b].) In particular, this means that $\mathsf{CLKID}^\omega(+\mathsf{PA})$ is *conservative* over CA. We reiterate that the interest behind the results of [BT17b] is rather the structural nature of the transformations, but this observation also exemplifies why CA is a natural and canonical object of study, as argued in [Sim17].

[9] For instance, Lemma 8.4 in that work yields a set of apparently exponential size in the worst case, and this bounds from below the size of the overall translation, e.g. as in Lemma 8.7.

10.4. **Cyclic propositional proof complexity.** One perspective gained from this work comes in the setting of *propositional proof complexity* (see, e.g., [CN10, Kra95]). Thm. 4.1 of Sect. 4 should relativise to theories with oracles too. For instance, we may formalise in $\mathrm{C}\Delta_0(f)$, where f is a fresh (uninterpreted) function symbol, a proof of the relativised version of the (finitary) pigeonhole principle (see App. A). This formula is known to be unprovable in $\mathrm{I}\Delta_0(f)$ due to lower bounds on propositional proofs of bounded depth [KPW95, PBI93].

At the same time the 'Paris-Wilkie' translation [PW81], which fundamentally links $\mathrm{I}\Delta_0(f)$ to bounded-depth propositional proofs, works locally on an arithmetic proof, at the level of formulae. Consequently one may still apply the translation to the lines of a $\mathrm{C}\Delta_0(f)$ proof to obtain small 'proof-like' objects containing only formulae of bounded depth, and a cyclic proof structure. One would expect that this corresponds to some strong form of 'extension', since it is known that adding usual extension to bounded systems already yields full 'extended Frege' proofs. However at the same time, some of this power has been devolved to the proof structure rather than simply at the level of the formula, and so could yield insights into how to prove simulations between fragments of Hilbert-Frege systems with extension.

We point out that recent work, [AL18], relating cyclic proof structures to proof complexity has already appeared, albeit with a different correctness criterion.

Acknowledgments

I am indebted to Alex Simpson for encouraging me to pursue this work and for his valuable feedback. Similarly, I would like to thank Stefano Berardi for several illuminating discussions on metalogical matters regarding cyclic proofs. Finally, I would like to thank James Brotherston, Guilhem Jaber, Alexis Saurin and the anonymous reviewers for this and previous versions of this work for all their helpful comments and insights.

References

[AL17] Bahareh Afshari and Graham E. Leigh. Cut-free completeness for modal μ-calculus. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*, pages 1–12, 2017.

[AL18] Albert Atserias and Massimo Lauria. Circular (yet sound) proofs. *CoRR*, abs/1802.05266, 2018.

[BBC08] James Brotherston, Richard Bornat, and Cristiano Calcagno. Cyclic proofs of program termination in separation logic. In *Proceedings of the 35th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2008, San Francisco, California, USA, January 7-12, 2008*, pages 101–112, 2008.

[BDP11] James Brotherston, Dino Distefano, and Rasmus Lerchedahl Petersen. Automated cyclic entailment proofs in separation logic. In *CADE-23 - 23rd International Conference on Automated Deduction, Wroclaw, Poland, July 31 - August 5, 2011. Proceedings*, pages 131–146, 2011.

[BDS16] David Baelde, Amina Doumane, and Alexis Saurin. Infinitary proof theory: the multiplicative additive case. In *25th EACSL Annual Conference on Computer Science Logic, CSL 2016, August 29 - September 1, 2016, Marseille, France*, pages 42:1–42:17, 2016.

[BGP12] James Brotherston, Nikos Gorogiannis, and Rasmus L. Petersen. A generic cyclic theorem prover. In *Programming Languages and Systems - 10th Asian Symposium, APLAS 2012, Kyoto, Japan, December 11-13, 2012. Proceedings*, pages 350–367, 2012.

[Bro05] James Brotherston. Cyclic proofs for first-order logic with inductive definitions. In *Automated Reasoning with Analytic Tableaux and Related Methods, International Conference, TABLEAUX 2005, Koblenz, Germany, September 14-17, 2005, Proceedings*, pages 78–92, 2005.

[Bro06] James Brotherston. *Sequent calculus proof systems for inductive definitions*. PhD thesis, University of Edinburgh, 2006.

[BS07] James Brotherston and Alex Simpson. Complete sequent calculi for induction and infinite descent. In *22nd IEEE Symposium on Logic in Computer Science (LICS 2007), 10-12 July 2007, Wroclaw, Poland, Proceedings*, pages 51–62, 2007.

[BS11] James Brotherston and Alex Simpson. Sequent calculi for induction and infinite descent. *J. Log. Comput.*, 21(6):1177–1216, 2011.

[BT17a] Stefano Berardi and Makoto Tatsuta. Classical system of Martin-Löf's inductive definitions is not equivalent to cyclic proof system. In *Foundations of Software Science and Computation Structures - 20th International Conference, FOSSACS 2017, Uppsala, Sweden, April 22-29, 2017, Proceedings*, pages 301–317, 2017.

[BT17b] Stefano Berardi and Makoto Tatsuta. Equivalence of inductive definitions and cyclic proofs under arithmetic. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*, pages 1–12, 2017.

[Bus95] Samuel R. Buss. The witness function method and provably recursive functions of Peano arithmetic. In *Studies in Logic and the Foundations of Mathematics*, volume 134, pages 29–68. Elsevier, 1995.

[Bus98] Samuel R. Buss, editor. *Handbook of Proof Theory*. Studies in Logic and the Foundations of Mathematics 137. Elsevier, 1998.

[CK02] Peter Clote and Evangelos Kranakis. *Boolean Functions and Computation Models*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2002.

[CN10] Stephen Cook and Phuong Nguyen. *Logical Foundations of Proof Complexity*. Cambridge University Press, New York, NY, USA, 1st edition, 2010.

[DBHS16] Amina Doumane, David Baelde, Lucca Hirschi, and Alexis Saurin. Towards completeness via proof search in the linear time μ-calculus: The case of Büchi inclusions. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '16, New York, NY, USA, July 5-8, 2016*, pages 377–386, 2016.

[DHL06] Christian Dax, Martin Hofmann, and Martin Lange. A proof system for the linear time μ-calculus. In *FSTTCS 2006: Foundations of Software Technology and Theoretical Computer Science, 26th International Conference, Kolkata, India, December 13-15, 2006, Proceedings*, pages 273–284, 2006.

[Dou17] Amina Doumane. Constructive completeness for the linear-time μ-calculus. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*, pages 1–12, 2017.

[DP17] Anupam Das and Damien Pous. A cut-free cyclic proof system for Kleene algebra. In *Automated Reasoning with Analytic Tableaux and Related Methods - 26th International Conference, TABLEAUX 2017, Brasília, Brazil, September 25-28, 2017, Proceedings*, pages 261–277, 2017.

[For14] Jérôme Fortier. *Puissance expressive des preuves circulaires. (Expressive Power of Circular Proofs)*. PhD thesis, Aix-Marseille University, Aix-en-Provence, France, 2014.

[FS13] Jérôme Fortier and Luigi Santocanale. Cuts for circular proofs: semantics and cut-elimination. In *Computer Science Logic 2013 (CSL 2013), September 2-5, 2013, Torino, Italy*, pages 248–262, 2013.

[Hir14] Denis R. Hirschfeldt. *Slicing the truth: On the computable and reverse mathematics of combinatorial principles*. World Scientific, 2014.

[HP93] Petr Hájek and Pavel Pudlák. *Metamathematics of First-Order Arithmetic*. Perspectives in mathematical logic. Springer, 1993.

[Kay91] Richard Kaye. *Models of Peano Arithmetic*. Oxford Logic Guides 15. Oxford University Press, 1991.

[KMPS19] Leszek Kołodziejczyk, Henryk Michalewski, Pierre Pradic, and Michał Skrzypczak. The logical strength of Büchi's decidability theorem. volume Volume 15, Issue 2, May 2019.

[KPW95] Jan Krajíček, Pavel Pudlák, and Alan Woods. An exponential lower bound to the size of bounded depth Frege proofs of the pigeonhole principle. *Random Structures & Algorithms*, 7(1):15–39, 1995.

[Kra95] Jan Krajíček. *Bounded arithmetic, propositional logic, and complexity theory*. Cambridge University Press, New York, NY, USA, 1995.

[McN66] Robert McNaughton. Testing and generating infinite sequences by a finite automaton. *Information and Control*, 9(5):521–530, 1966.

[ML71] Per Martin-Löf. Hauptsatz for the intuitionistic theory of iterated inductive definitions. In J.E. Fenstad, editor, *Proceedings of the Second Scandinavian Logic Symposium*, volume 63 of *Studies in Logic and the Foundations of Mathematics*, pages 179 – 216. Elsevier, 1971.

[NST18] Rémi Nollet, Alexis Saurin, and Christine Tasson. Local validity for circular proofs in linear logic with fixed points. In *27th EACSL Annual Conference on Computer Science Logic, CSL 2018, September 4-7, 2018, Birmingham, UK*, pages 35:1–35:23, 2018.

[NW96] Damian Niwinski and Igor Walukiewicz. Games for the μ-calculus. *Theor. Comput. Sci.*, 163(1&2):99–116, 1996.

[Par71] Rohit Parikh. Existence and feasibility in arithmetic. *J. Symb. Log.*, 36(3):494–508, 1971.

[Par72] Charles Parsons. On n-quantifier induction. *The Journal of Symbolic Logic*, 37(3):466–482, 1972.

[PBI93] Toniann Pitassi, Paul Beame, and Russell Impagliazzo. Exponential lower bounds for the pigeonhole principle. *Computational Complexity*, 3:97–140, 1993. 10.1007/BF01200117.

[PW81] Jeff B. Paris and Alex J. Wilkie. Δ_0 sets and induction. *Open Days in Model Theory and Set Theory, W. Guzicki, W. Marek, A. Pelc, and C. Rauszer, eds*, pages 237–248, 1981.

[RB17] Reuben N. S. Rowe and James Brotherston. Automatic cyclic termination proofs for recursive procedures in separation logic. In *Proceedings of the 6th ACM SIGPLAN Conference on Certified Programs and Proofs, CPP 2017, Paris, France, January 16-17, 2017*, pages 53–65, 2017.

[San02] Luigi Santocanale. A calculus of circular proofs and its categorical semantics. In *Foundations of Software Science and Computation Structures, 5th International Conference, FOSSACS 2002, Grenoble, France, April 8-12, 2002, Proceedings*, pages 357–371, 2002.

[Sch77] Kurt Schütte. *Proof Theory*. Grundlehren der mathematischen Wissenschaften 225. Springer Berlin Heidelberg, 1977. Translation of Beweistheorie, 1968.

[SD03] Christoph Sprenger and Mads Dam. On the structure of inductive reasoning: Circular and tree-shaped proofs in the μ-calculus. In *Foundations of Software Science and Computational Structures, 6th International Conference, FOSSACS 2003, Warsaw, Poland, April 7-11, 2003, Proceedings*, pages 425–440, 2003.

[Sim09] Stephen G. Simpson. *Subsystems of second order arithmetic*, volume 1. Cambridge University Press, 2009.

[Sim17] Alex Simpson. Cyclic arithmetic is equivalent to Peano arithmetic. In *Foundations of Software Science and Computation Structures - 20th International Conference, FOSSACS 2017, Proceedings*, pages 283–300, 2017.

[Str17] Sorin Stratulat. Cyclic proofs with ordering constraints. In *Automated Reasoning with Analytic Tableaux and Related Methods - 26th International Conference, TABLEAUX 2017, Brasília, Brazil, September 25-28, 2017, Proceedings*, pages 311–327, 2017.

[Tho97] Wolfgang Thomas. Languages, automata, and logic. In *Handbook of formal languages*, pages 389–455. Springer, 1997.

Appendix A. Case study: the relativised pigeonhole principle

In this section we will give an example of the translation from Sect. 4 in a relativised setting. Simpson already gave an example of a separation between $\mathrm{C}\Sigma_1$ and $\mathrm{I}\Sigma_1$ via the totality of the Ackermann-Péter function [Sim17], a Π_2 sentence. Logically simpler Π_1 separations are obtainable in the form of consistency principles, as we discussed in Sect. 8.

In this section we consider the well-known *pigeonhole principle*, defined by the following FO formula with an uninterpreted function symbol f:

$$\mathsf{PHP}(f) \;:=\; \forall n.(\forall x \leq n.f(x) < n \;\supset\; \exists x \leq n.\exists x' < x.f(x) = f(x'))$$

It is well-known that $\mathrm{I}\Delta_0(f)$ does not prove $\mathsf{PHP}(f)$, due to lower bounds on propositional proofs of bounded depth [KPW95, PBI93]. On the other hand, by relativising the constructions in Sect. 4, it is provable in $\mathrm{C}\Delta_0(f)$ thanks to the known simple proofs in $\mathrm{I}\Sigma_1(f)$.

A.1. **A simple proof of $\mathsf{PHP}(f)$ in $\mathrm{I}\Sigma_1(f)$.** First we recall a simple well-known proof of $\mathsf{PHP}(f)$ in $\mathrm{I}\Sigma_1(X)$. In fact, as in Sect. 4, we will work with Π_1-IND rather than Σ_1-IND.

Temporarily, let us write A, B for first-order variables that we interpret as the codes of finite sets. For such codes we may use set-theoretic symbols such as $\in$ and $\setminus$ with their usual interpretations, with the understanding that their basic properties are provable in $\mathrm{I}\Delta_0$.

Lemma A.1. $\mathrm{I}\Sigma_1(f)$ *proves the following:*

$$\forall A, B.(|A| > |B| \supset (\forall x \in A.f(x) \in B \supset \exists x, x' \in A.(x \neq x' \wedge f(x) = f(x')))) \tag{A.1}$$

Proof. Working in $\mathrm{I}\Sigma_1(f)$, we reason by induction on $|B|$. If B is empty and $|A| > |B|$ then A is nonempty and so (A.1) is vacuously true by falsity of the premiss.

Otherwise B is nonempty, so let $b \in B$ and let $|A| > |B|$.

- If $\exists x \in A.f(x) = b$ then, let $a \in A$ such that $f(a) = b$.
  - If $\exists x \in A.(x' \neq a \wedge f(x') = b)$ then we are done.
  - Otherwise suppose $\forall x \in A.(f(x) = b \supset x = a)$. Then we have $\forall x \in A \setminus \{a\}.f(x) \in B \setminus \{b\}$. Since we still have that $|A \setminus \{a\}| > |B \setminus \{b\}$ we may conclude by the inductive hypothesis.
- Otherwise $\forall x \in A.f(x) \neq b$, so in fact $\forall x \in A.f(x) \in B \setminus \{b\}$ and still $|A| > |B \setminus \{b\}$. Hence we conclude by the inductive hypothesis. □

From here there is a simple proof of (A.1) $\supset \mathsf{PHP}(f)$ in $\mathrm{I}\Delta_0(f)$, by instantiating A and B as $[0, n]$ and $[0, n)$ resp. Thus we have that $\mathrm{I}\Sigma_1(f) \vdash \mathsf{PHP}(f)$.

A.2. **A proof of $\mathsf{PHP}(f)$ in $\mathrm{C}\Delta_0(f)$.** To show that $\mathrm{C}\Delta_0(f) \vdash \mathsf{PHP}(f)$ it suffices to give $\mathrm{C}\Delta_0(f)$ proofs of Lemma A.1. The remainder of the argument may be carried out in $\mathrm{I}\Delta_0(f)$, and so also in $\mathrm{C}\Delta_0(f)$ by Prop. 3.7,

Lemma A.2. $\mathrm{C}\Delta_0(f) \vdash$ (A.1).

Proof. As abbreviations, let us write $f(A) \subseteq B$ for $\forall x \in A.f(x) \in B$ and $\mathrm{Inj}_f(A)$ for $\forall x, x' \in A.(f(x) = f(x') \supset x = x')$. We give an appropriate derivation in Fig. 2, mimicking the argument of Lemma A.1 under Lemma 4.2, where π_0 is a $\mathrm{I}\Delta_0(f)$ proof of $B = \varnothing, |A| > |B|, f(A) \subseteq B \Rightarrow \neg\mathrm{Inj}_f(A)$ and π_1 is a proof of $a' \in A, a' \neq a, f(a') = b, a \in A, f(a) = b, b \in B, |A| > |B|, f(A) \subseteq B \Rightarrow \neg\mathrm{Inj}_f(A)$. □

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdots}{|A| > |B|, f(A) \subseteq B \Rightarrow \neg\mathrm{Inj}_f(A)}\bullet}{sub\ \dfrac{}{|A \setminus \{a\}| > |B \setminus \{b\}|, f(A \setminus \{a\}) \subseteq B \setminus \{b\} \Rightarrow \neg\mathrm{Inj}_f(A \setminus \{a\})}}}{\forall x \in A.(f(x) = b \supset x = a), a \in A, f(a) = b, |A| > |B|, f(A) \subseteq B \Rightarrow \neg\mathrm{Inj}_f(A)} \quad \pi_1}{a \in A, f(a) = b, b \in B, |A| > |B|, f(A) \subseteq B \Rightarrow \neg\mathrm{Inj}_f(A)} \quad \dfrac{\dfrac{\dfrac{\vdots}{|A| > |B|, f(A) \subseteq B \Rightarrow \neg\mathrm{Inj}_f(A)}\bullet}{sub\ \dfrac{}{|A| > |B \setminus \{b\}|, f(A) \subseteq B \setminus \{b\} \Rightarrow \neg\mathrm{Inj}_f(A)}}}{\forall x \in A.\neg f(x) = b, b \in B, |A| > |B|, f(A) \subseteq B \Rightarrow \neg\mathrm{Inj}_f(A)}}{b \in B, |A| > |B|, f(A) \subseteq B \Rightarrow \neg\mathrm{Inj}_f(A)}\bullet}{\supset,\forall\ \dfrac{|A| > |B|, f(A) \subseteq B \Rightarrow \neg\mathrm{Inj}_f(A)}{\Rightarrow \forall A, B.(|A| > |B| \supset (f(A) \subseteq B \supset \neg\mathrm{Inj}_f(A)))}} \quad \pi_0$$

Figure 2: A $\mathrm{C}\Delta_0(f)$ proof of $\mathsf{PHP}(f)$, where π_0 is a $\mathrm{I}\Delta_0(f)$ proof of $B = \varnothing, |A| > |B|, f(A) \subseteq B \Rightarrow \neg\mathrm{Inj}_f(A)$ and π_1 is a proof of $a' \in A, a' \neq a, f(a') = b, a \in A, f(a) = b, |A| > |B|, f(A) \subseteq B \Rightarrow \neg\mathrm{Inj}_f(A)$.

Appendix B. Non-uniform complementation of Büchi automata

In this section we give a self-contained proof of Prop. 7.1, which is only implicit in [KMPS19]. One novel contribution here is a much simpler proof of the (Nonuniform) Additive Ramsey Theorem. The remainder of the complementation argument is standard and follows closely [KMPS19], though we present it here with more structure and proof details.

B.1. **Nonuniform Additive Ramsey Theorem in RCA_0.** Let us write $\binom{\mathbb{N}}{2}$ for the set of unordered pairs of natural numbers. We in fact write its elements as ordered pairs (i, j) where always $i < j$.

Let $(S, \bullet)$ be a finite semigroup and consider a 'colouring' $C : \binom{\mathbb{N}}{2} \to S$. We may omit the group operation symbol $\bullet$ when composing elements of S. We say that C is *additive* if, whenever $i < j < k$, we have $C(i, j)C(j, k) = C(i, k)$. We say that $I \subseteq \mathbb{N}$ is an *a-clique* (under C) if, $\forall i, j \in I$ we have $C(i, j) = a$.

Theorem B.1 (Nonuniform Additive Ramsey Theorem)**.** *Let $(S, \bullet)$ be a finite semigroup. Then:*

$$\mathsf{RCA}_0 \vdash \forall C : \binom{\mathbb{N}}{2} \to S.(\text{“}C \text{ is additive”} \supset \exists a \in S.\exists I \subseteq \mathbb{N}.\text{“}I \text{ is an infinite } a\text{-clique”})$$

Before we give the proof, we better state the following fact:

Fact B.2 (Nonuniform Infinite Pigeonhole Principle)**.** Let S be a finite set. Then:

$$\mathsf{RCA}_0 \vdash \forall f : \mathbb{N} \to S.\exists a \in S.\forall m.\exists x > m.f(x) = a$$

In particular:

$$\mathsf{RCA}_0 \vdash \forall f : \mathbb{N} \to S.\ \exists Y \text{infinite}.\ \forall x, y \in Y.\ f(x) = f(y)$$

This result is well-known in reverse mathematics and can be proved by a routine meta-level induction on the size of S. The second display follows from the first part by Δ^0_1-CA.

We can now give a proof of the Nonuniform Additive Ramsey Theorem. This argument differs from and is somewhat simpler than the analogous argument from [KMPS19] (Prop. 4.1), which requires a detour via the Ordered Ramsey Theorem and 'Green' theory. In particular, since we only care about nonuniform provability here, we are not conerned about the quantifier complexity of the inductive invariant, since this induction will take place at the meta-level.

Proof of Thm. B.1. We proceed by a meta-level induction on the size of S. If S is empty the statement is vacuously true, so we proceed to the inductive step.

If for some $a \in S$ there are only finitely many i such that $C(i, j) = a$ for some $j > i$. Then, letting n be such that $C(i, j) \neq a$ for $i, j \geq n$, we may simply apply the inductive hypothesis to the colouring $C(\cdot + n, \cdot + n)$. Thus we may assume henceforth that:

$$\forall a \in S.\exists^\infty i \in \mathbb{N}.\exists j > i.C(i, j) = a \tag{B.1}$$

Now, suppose that for some $a \in S$ we have $aS \subsetneq S$. We will define, assuming the above display, a certain 'increasing' enumeration of elements of $\binom{\mathbb{N}}{2}$ that map to a. Consider the

functions $i : \mathbb{N} \to \mathbb{N}$ and $j : \mathbb{N} \to \mathbb{N}$ defined simultaneously as follows:

$$\begin{array}{rcl} i(0) & := & \beta_0 \mu k.(k = \langle i', j' \rangle \wedge j' > i' \wedge C(i', j') = a) \\ i(n+1) & := & \beta_0 \mu k.(k = \langle i', j' \rangle \wedge j' > i' > j(n) \wedge C(i', j') = a) \\ j(0) & := & \mu j' > i(0).C(i(0), j') = a \\ j(n+1) & := & \mu j' > i(n+1).C(i(n+1), j') = a \end{array}$$

where we write $\mu x.\varphi(x)$ for the least x such that $\varphi(x)$. Notice that this definition is a form of simultaneous primitive recursion with provably terminating blind searches, assuming Eqn. (B.1), so i and j are provably recursive and their defining equations above are provable (in $\mathsf{RCA}_0 + (\text{B.1})$). Thus we have that for all $n \in \mathbb{N}$:

$$\begin{array}{l} i(n) < j(n) < i(n+1) \\ C(i(n), j(n)) = a \end{array}$$

But now, for $m < n$, notice that,

$$\begin{array}{rcl} C(i(m), i(n)) & = & C(i(m), j(m))C(j(m), i(n)) \\ & = & aC(j(m), i(n)) \\ & \in & aS \subsetneq S \end{array}$$

by the additivity property and assumption. So from here we may apply the inductive hypothesis to the colouring $C(i(\cdot), i(\cdot))$ and conclude. Thus we may henceforth assume that:

$$\forall a \in S.aS = S \tag{B.2}$$

In other words, $a \bullet \cdot$ is a bijection on S, for any $a \in S$. Thus we have the left cancellation property: if $ab = ac$ then $b = c$, and for any a there is a unique $b \in S$ such that $ab = a$.

Now, $C(0, \cdot)$ must take some value $a \in S$ infinitely often, by the non-uniform infinite pigeonhole principle, Fact B.2. In this case we may simply set $I = \{i : C(0, i) = a\}$. Notice that, for $i, j \in I$ with $i < j$, we have $a = C(0, j) = C(0, i)C(i, j) = aC(i, j)$, and so indeed each $C(i, j)$ is identical, by left-cancellation. □

B.2. **Characterising rejection via the Ramseyan factorisation of ω-words.** For the remainder of this section let us fix an NBA $\mathcal{A} = (A, Q, \delta, q_0, F)$.

Since we deal with non-deterministic automata, it no longer makes sense to use the notation $\sigma : q \xrightarrow[\delta]{*} q'$ from Sect. 9, since there may be several paths through δ from q to q' following σ. Instead, for $\sigma \in A^*$, we write:

$$x : q \xrightarrow[\delta]{\sigma} q' \quad := \quad \text{“}x \text{ is a path through } \delta \text{ from } q \text{ to } q' \text{ following } \sigma\text{”}$$

Now, for each finite word $\sigma \in A^*$ we may consider its *transition matrix* $\delta(\sigma)$, which is the graph whose nodes are those of Q with an edge from q to q' if there exists some $x : q \xrightarrow[\delta]{\sigma} q'$. The edge is labelled with ∞ if there is such a path that hits an accepting state (after q). We construe such graphs as functions of type $Q \times Q \to \{0, 1, \infty\}$ in the natural way. Let us call the set of all such graphs $\delta(A^*)$, which we note must be finite. Notice that $\delta : A^* \to (Q \times Q \to \{0, 1, \infty\})$ is provably recursive in RCA_0 by the following definition:

$$\delta(\sigma)(q, q') = \begin{cases} 0 & \nexists x : q \xrightarrow[\delta]{\sigma} q' \\ \infty & \exists x : q \xrightarrow[\delta]{\sigma} q'.\exists i < |x|.x(i) \in F \\ 1 & \text{otherwise} \end{cases}$$

We may compose such graphs by a variation of the usual relational composition accounting for the labelling in a natural way: for $\beta, \gamma : Q \times Q \to \{0, 1, \infty\}$, we define $\beta\gamma : Q \times Q \to \{0, 1, \infty\}$ by:

$$(\beta\gamma)(q, q') = \max_{p \in Q} \big(\beta(q, p) \cdot \gamma(p, q')\big)$$

Here we define max and $\cdot$ as expected, in particular setting $0 \cdot \infty = 0 = \infty \cdot 0$. Intuitively, this is just the same as relational composition, only recording if it is possible to hit a final state en route. Under these operations we have that δ is in fact a homomorphism $A^* \to (Q \times Q \to \{0, 1, \infty\})$, provably in RCA_0:

Proposition B.3. $\mathsf{RCA}_0 \vdash \forall \sigma, \tau \in A^*.\delta(\sigma\tau) = \delta(\sigma)\delta(\tau)$.

The proof follows by a straightforward induction on the length of τ. Only the base case when τ is some $a \in A$ is interesting, with the inductive case following by associativity of word composition.

Now, for $\beta, \gamma : Q \times Q \to \{0, 1, \infty\}$, let us say that the pair (β, γ) is *rejecting* if:

(1) $\beta\gamma = \beta$; and,
(2) $\gamma\gamma = \gamma$; and,
(3) $\forall q \in Q.(\beta(q_0, q) > 0 \supset \gamma(q, q) < \infty)$

Again, the property of being rejecting pair is clearly Δ^0_1, since we have fixed Q in advance so there are only finitely many cases to consider. Let us adopt the interval notation $[i, j]$ for the set $\{i, i + 1, \ldots, j\}$ and $[i, j)$ for the set $\{i, i + 1, \ldots, j - 1\}$. For an infinite sequence X we also write $X[i, j]$ and $X[i, j)$ for the finite subsequences $(X(i), X(i + 1), \ldots, X(j))$ and $(X(i), X(i + 1), \ldots, X(j - 1))$ respectively.

Lemma B.4 (Ramseyan factorisation (in RCA_0))**.** *For any $X \in A^\omega$, there are $\beta, \gamma \in \delta(A^*)$ and an infinite set $I \subseteq \mathbb{N}$ such that:*

(1) *$\delta(X[0, i)) = \beta$, for $i \in I$.*
(2) *$\delta(X[i, j)) = \gamma$, for $i, j \in I$ with $i < j$.*

Moreover, for any such β, γ, I satisfiying (1) *and* (2) *above, we have that:*

(3) *(β, γ) is a rejecting pair if and only if $X \notin \mathcal{L}(\mathcal{A})$.*

Proof. Working inside RCA_0, let $C : \binom{\mathbb{N}}{2} \to \delta(A^*)$ by $C(i, j) = \delta(X[i, j))$. By Prop. B.3 we have that C is an additive colouring, and thus we may apply the Nonuniform Additive Ramsey Theorem, Thm. B.1, to obtain some infinite set I_0 with $\gamma = C(i, j)$, for all $i, j \in I_0$ with $i < j$, yielding (2). Now, for an arbitrary $i_0 \in I_0$ we may set $I = \{i \in I_0 : i > i_0\}$. Notice that, now, $C(0, i) = C(0, i_0)C(i_0, i) = C(0, i_0)\gamma$, so we may set $\beta = C(0, i)$ for some/any $i \in I$, yielding (1).

For (3), first suppose (β, γ) is a rejecting pair and let $Y \in Q^\omega$ be a run of $\mathcal{A}$ on X. We will show that Y cannot be accepting. By the Non-Uniform Infinite Pigeonhole Principle, Fact B.2, there is some $q \in Q$ and some infinite subset $I' \subseteq I$ such that $Y(i) = q$ for each $i \in I'$. We claim that, for any $i, j \in I'$ and $k \in \mathbb{N}$ such that $i < k < j$, we have that $q_k \notin F$. For this notice that:

- $\beta(q_0, q) > 0$ since $Y[0, i] : q_0 \xrightarrow[\delta]{X[0,i)} q$.
- $\gamma(q, q) < \infty$ since (β, γ) is a rejecting pair.

So, if q_k lies on the path $Y[i, j] : q \xrightarrow[\delta]{X[i,j)} q$, we must have that $q_k \notin F$, appealing to Prop. B.3. Thus, for any $i \in I'$, we have that $\forall k > i.Y(k) \notin F$, and so Y is not an accepting run.

Now, suppose that (β, γ) is not a rejecting pair and, by definition, let q be such that $\beta(q_0, q) > 0$ and $\gamma(q, q) = \infty$. We may enumerate infinite sets in RCA_0 so let $I = (i_j)_{j\in\mathbb{N}}$. We may now simply define an accepting run Y of $\mathcal{A}$ on X by recursive comprehension by insisting that,

- in the interval $[0, i_0)$, Y follows the 'least' path through δ from q_0 to q; and,
- in the interval $[i_j, i_{j+1})$, Y follows the 'least' path through δ from q to q hitting a final state.

Such paths must exist since (β, γ) is not a rejecting pair, and the set of all paths of bounded length may be enumerated in RCA_0. By construction, Y is accepting. □

B.3. **The complement NBA and proof of correctness.** Now we are ready to define the complement automaton of $\mathcal{A}$, which simply guesses a rejecting Ramseyan factorisation of an input ω-word:

Definition B.5. We define the NBA $\mathcal{A}^c = (A, Q^c, \delta^c, q_0^c, F^c)$ as follows:

- $Q^c = \{q_0\} \cup \delta(A^*) \cup \delta(A^*)^2 \cup \delta(A^*)^3$.
- δ^c consists of the following transitions:
  - $(q_0, a, (\beta, \gamma, \delta(a))$, for each rejecting pair (β, γ).
  - $((\beta, \gamma, \zeta), a, (\beta, \gamma, \zeta\delta(a)))$
  - $((\beta, \gamma, \zeta), a, \gamma)$ if $\zeta\delta(a) = \beta$.
  - $(\gamma, a, (\gamma, \delta(a)))$.
  - $((\gamma, \zeta), a, (\gamma, \zeta\delta(a)))$.
  - $((\gamma, \zeta), a, \gamma)$ if $\zeta\delta(a) = \gamma$.
  - (γ, a, γ) if $\delta(a) = \gamma$.
- $q_0^c = q_0$.
- $F^c = \delta(A^*)$.

Now we are ready to prove the non-uniform complementation result.

Proof of Prop. 7.1. By Lemma B.4, it suffices to show that $\mathcal{A}^c$ accepts an ω-word just if has a Ramseyan factorisation that is rejecting. Let us proceed working inside RCA_0.

First, suppose $X \in A^\omega$ with $X \notin \mathcal{L}(\mathcal{A})$ and let β, γ, I be obtained from Lemma B.4. Again we may enumerate $I = (i_j)_{j\in\mathbb{N}}$. Define the run Y of X on $\mathcal{A}^c$ as follows:

- $Y(0) = q_0$ and $Y(1) = (\beta, \gamma, \delta(X(0)))$;
- in the interval $[1, i_0)$, Y follows the (unique) transitions of the form $((\beta, \gamma, \zeta), a, (\beta, \gamma, \zeta\delta(a)))$;
- $Y(i_j) = \gamma$, for all $j \in \mathbb{N}$;
- in an interval (i_j, i_{j+1}), Y follows the (unique) transitions of the form $((\gamma, \zeta), a, (\gamma, \zeta\delta(a)))$.

Y is clearly recursive and so is indeed definable by Δ^0_1-CA. Moreover, Y hits the final state γ infinitely often (at each $i_j \in I$), so Y is an accepting run for X on $\mathcal{A}^c$.

Conversely, suppose $X \in \mathcal{L}(\mathcal{A}^c)$ and let Y be an accepting run with $Y(1) = (\beta, \gamma, \delta(X(0)))$. By a routine induction, Y must hit γ infinitely often, since that is the only state in $\delta(A^*)$ that can ever be hit. Thus, by Δ^0_1-CA, we have an infinite set $I = \{i \in \mathbb{N} : Y(i) = \gamma\}$, which we again enumerate $I = (i_j)_{j\in\mathbb{N}}$. We may now show the properties (1) and (2) of Lemma B.4 with respect to the β, γ, I thus defined by induction. More precisely, we prove by induction on $i \in \mathbb{N}$ that $Y(i)$ has the following form:

- q_0, if $i = 0$;
- $(\beta, \gamma, \delta(X[0, i)))$, if $i < i_0$;

- γ, if $i \in I$;
- $(\gamma, \delta(X[i_j, i]))$ if $i_j < i < i_{j+1}$.

Properties (1) and (2) from Lemma B.4 now follow as special cases. Since (β, γ) was a rejecting pair, by definition of $\mathcal{A}^c$, we have from Lemma B.4.(3) that $\mathcal{A}$ rejects X. This concludes the proof. □