On the logical complexity of cyclic arithmetic

Anupam Das

University of Birmingham

Moscow Proof Theory Seminar
23rd March 2020
(via video-conference)

1Partially supported by a Marie Skłodowska-Curie fellowship, ERC project 753431.
Irrationality of $\sqrt{2}$ via infinite descent

Consider the following 'derivation' over $\mathbb{N}^+$:

$\Rightarrow \sqrt{2}$ is prime

$\Rightarrow b\sqrt{2} = \sqrt{2}c \Rightarrow c < a$, $c\sqrt{2} = b\sqrt{2} \Rightarrow \exists x < a, a\sqrt{2} = b\sqrt{2} \Rightarrow \Rightarrow \forall x, y, x\sqrt{2} \neq y\sqrt{2}$

Apparently non-wellfounded reasoning.

Why is it sound?
Irrationality of $\sqrt{2}$ via infinite descent

Consider the following ‘derivation’ over $\mathbb{N}^+$:

\[
\begin{align*}
\vdots \\
\frac{b^2 = 2c^2}{b^2 = 2c^2} \Rightarrow \\
\frac{c < a, \ 4c^2 = 2b^2}{c < a, \ 4c^2 = 2b^2} \Rightarrow \\
\Rightarrow 2 \text{ is prime} & \quad \exists x < a. a = 2x, \ a^2 = 2b^2 \Rightarrow \\
\frac{a^2 = 2b^2}{a^2 = 2b^2} \Rightarrow \\
\Rightarrow \forall x, y. x^2 \neq 2y^2
\end{align*}
\]
Irrationality of $\sqrt{2}$ via infinite descent

Consider the following ‘derivation’ over $\mathbb{N}^+$:

\[
\begin{align*}
\therefore & b^2 = 2c^2 \Rightarrow c < a, 4c^2 = 2b^2 \Rightarrow \\
\Rightarrow & 2 \text{ is prime} \quad \exists x < a. a = 2x, a^2 = 2b^2 \Rightarrow \\
\therefore & a^2 = 2b^2 \Rightarrow \\
\Rightarrow & \forall x, y. x^2 \neq 2y^2
\end{align*}
\]

• Apparently non-wellfounded reasoning.
Consider the following ‘derivation’ over \( \mathbb{N}^+ \):

\[
\begin{align*}
\vdots \\
&b^2 = 2c^2 \implies \quad \bullet \\
&c < a, \ 4c^2 = 2b^2 \implies \\
&\Rightarrow 2 \text{ is prime} \quad \exists x < a. a = 2x, a^2 = 2b^2 \implies \\
&\quad a^2 = 2b^2 \implies \\
&\quad \Rightarrow \forall x, y. x^2 \neq 2y^2
\end{align*}
\]

- Apparently non-wellfounded reasoning.
- Why is it sound?
1. Peano and Cyclic Arithmetic

2. Summary of previous work and contributions

3. From induction to cycles

4. From cycles to induction

5. Some further results

6. Conclusions
Cyclic proofs

- Proof theory for FOL with inductive definitions.
- (Automated) proofs of program termination in separation logic.
- Proof systems for the modal $\mu$-calculus and other fixed point logics.
- Type systems based on fragments of linear logic with fixed points.
- Metalogical results, like interpolation.
- Proof search procedures.

A motivating abstract question:

Question (Brotherston-Simpson conjecture): Are inductive proofs and cyclic proofs equally powerful? This talk is about the special case of first-order arithmetic.
Cyclic proofs

- Proof theory for FOL with inductive definitions.
- (Automated) proofs of program termination in separation logic.
- Proof systems for the modal $\mu$-calculus and other fixed point logics.
- Type systems based on fragments of linear logic with fixed points.
- Metalogical results, like interpolation.
- Proof search procedures.
- ...
Cyclic proofs

- Proof theory for FOL with inductive definitions.
- (Automated) proofs of program termination in separation logic.
- Proof systems for the modal $\mu$-calculus and other fixed point logics.
- Type systems based on fragments of linear logic with fixed points.
- Metalogical results, like interpolation.
- Proof search procedures.
- ...

A motivating abstract question:

**Question (Brotherston-Simpson conjecture)**

Are inductive proofs and cyclic proofs equally powerful?
Cyclic proofs

- Proof theory for FOL with inductive definitions.
- (Automated) proofs of program termination in separation logic.
- Proof systems for the modal $\mu$-calculus and other fixed point logics.
- Type systems based on fragments of linear logic with fixed points.
- Metalogical results, like interpolation.
- Proof search procedures.
- ...

A motivating abstract question:

**Question (Brotherston-Simpson conjecture)**

Are inductive proofs and cyclic proofs equally powerful?

This talk is about the special case of **first-order arithmetic**.
A sequent calculus presentation of \( \text{PA} \)

\( \Delta / \text{zero.lf} \) - initial sequents for the instances of \( \mathcal{Q} \): defining properties of \( \text{zero.lf} \), \( \text{s} \), \( \text{+} \), \( \times \), \( < \).

- An induction rule:
  \[
  \Gamma \Rightarrow \Delta, A(\text{zero.lf}) \quad \Gamma, A(sa) \Rightarrow \Delta,
  \]

- We include an explicit substitution rule for unifying sequents in cycles:
  \[
  \Gamma \Rightarrow \Delta, \theta \quad \Gamma \theta \Rightarrow \Delta \theta.
  \]

Definition
\( \Phi \) is the fragment of \( \text{PA} \) where induction is restricted to formulae \( A \in \Phi \). In particular \( \text{I} \Sigma_n \) has induction only on formulae \( \exists x / \text{one.lf} \), \( \forall x / \text{two.lf} \), \( \ldots \), \( Qx_n \).

A recursive.
**Peano Arithmetic**, written PA, can be specified by a deduction system as follows:

- **Δ₀-initial sequents** for the instances of Q: defining properties of 0, s, +, ×, <.
- **An induction rule:**

\[
\begin{align*}
\Gamma &\Rightarrow \Delta, A(0) \quad \Gamma, A(a) \Rightarrow \Delta, A(sa) \\
\hline
\Gamma &\Rightarrow \Delta, A(t)
\end{align*}
\]
Peano Arithmetic, written PA, can be specified by a deduction system as follows:

- **$\Delta_0$-initial sequents** for the instances of $Q$: defining properties of $0$, $s$, $+$, $\times$, $\lt$.
- An induction rule:

$$
\begin{align*}
\Gamma \Rightarrow \Delta, A(0) & & \Gamma, A(a) \Rightarrow \Delta, A(sa) \\
\hline
\Gamma \Rightarrow \Delta, A(t)
\end{align*}
$$

- We include an explicit substitution rule for unifying sequents in cycles:

$$
\theta\text{-sub} \quad \frac{\Gamma \Rightarrow \Delta}{\theta(\Gamma) \Rightarrow \theta(\Delta)}
$$
A sequent calculus presentation of PA

**Peano Arithmetic**, written PA, can be specified by a deduction system as follows:

- **Δ₀-initial sequents** for the instances of Q: defining properties of 0, s, +, ×, <.
- **An induction rule:**
  \[
  \frac{\Gamma \Rightarrow \Delta, A(0) \quad \Gamma, A(a) \Rightarrow \Delta, A(sa)}{\Gamma \Rightarrow \Delta, A(t)}
  \]
- We include an explicit **substitution rule** for unifying sequents in cycles:
  \[
  \theta\text{-sub } \frac{\Gamma \Rightarrow \Delta}{\theta(\Gamma) \Rightarrow \theta(\Delta)}
  \]

**Definition**

*Φ* is the fragment of PA where induction is restricted to formulae \( A \in \Phi \). In particular *ΙΣₙ* has induction only on formulae \( \exists x_1. \forall x_2. \ldots Qx_n.A \), with A **recursive**.
Proposition (Folklore)

For $n \geq 0$ we have that $I\Sigma_n = I\Pi_n$. 
Some proof theory of arithmetic

Proposition (Folklore)
For $n \geq 0$ we have that $I\Sigma_n = I\Pi_n$.

Theorem ((Free-)cut elimination)
If $\text{PA} \vdash S(\bar{a})$, then there is a sequent proof $\pi$ of $S(\bar{a})$ containing only subformulae of $S(\bar{a})$, an induction formula of $\pi$ or an initial sequent of $\pi$. 
Some proof theory of arithmetic

Proposition (Folklore)
For $n \geq 0$ we have that $I\Sigma_n = I\Pi_n$.

Theorem ((Free-)cut elimination)
If $\text{PA} \vdash S(\bar{a})$, then there is a sequent proof $\pi$ of $S(\bar{a})$ containing only subformulae of $S(\bar{a})$, an induction formula of $\pi$ or an initial sequent of $\pi$.

Corollary
For $n \geq 0$, if $I\Sigma_n \vdash \forall \bar{x}. \varphi(\bar{x})$, for $\varphi \in \Sigma_n$, then $\Rightarrow \varphi(\bar{a})$ has a sequent proof containing only $\Sigma_n$ formulae.
Non-wellfounded arithmetic (Simpson ’12)

A preproof is a locally correct infinite derivation tree. Let \((S_i)\) be an infinite branch of a preproof. We say \(t'\) is a precursor of \(t\) at \(i\) if:

- \(S_i\) concludes a \(\theta\)-sub-step and \(t = \theta(t')\);
- \(S_i\) concludes any other step and \(t'\) is \(t\);
- \(S_i\) concludes any other step and \(t = t'\) occurs in the antecedent of \(S_i\).

A trace along an infinite branch \((S_i)\) is a sequence \((t_i)\) such that:

- \(t_i\) is a a precursor of \(t_{i+1}\);
- \(t_{i+1}\) occurs in the antecedent of \(S_i\). (a 'progress point')

Definition (\(\infty\)-proofs)

A \(\infty\)-proof (or just 'proof') is a preproof where each infinite branch has an infinitely progressing trace.
Definition (Precursors and traces)
A preproof is a locally correct infinite derivation tree.
Definition (Precursors and traces)
A preproof is a locally correct infinite derivation tree. Let \((S_i)_i\) be an infinite branch of a preproof. We say \(t'\) is a precursor of \(t\) at \(i\) if:

- \(S_i\) concludes a \(\theta\)-sub step and \(t = \theta(t')\); or
- \(S_i\) concludes any other step and \(t'\) is \(t\); or
- \(S_i\) concludes any other step and \(t = t'\) occurs in the antecedent of \(S_i\).
Definition (Precursors and traces)

A \textbf{preproof} is a locally correct infinite derivation tree. Let \((S_i)_i\) be an infinite branch of a preproof. We say \(t'\) is a \textbf{precursor} of \(t\) at \(i\) if:

- \(S_i\) concludes a \(\theta\)-sub step and \(t = \theta(t')\); or
- \(S_i\) concludes any other step and \(t'\) is \(t\); or
- \(S_i\) concludes any other step and \(t = t'\) occurs in the antecedent of \(S_i\).

A \textbf{trace} along an infinite branch \((S_i)_i\) is a sequence \((t_i)_{i \geq n}\) such that:

1. \(t_i\) is a precursor of \(t_{i+1}\); or
2. \(t_{i+1} < t_i\) occurs in the antecedent of \(S_i\). (a ‘progress point’)

Non-wellfounded arithmetic (Simpson ’12)
Definition (Precursors and traces)
A preproof is a locally correct infinite derivation tree. Let \((S_i)_i\) be an infinite branch of a preproof. We say \(t'\) is a precursor of \(t\) at \(i\) if:

- \(S_i\) concludes a \(\theta\)-sub step and \(t = \theta(t')\); or
- \(S_i\) concludes any other step and \(t'\) is \(t\); or
- \(S_i\) concludes any other step and \(t = t'\) occurs in the antecedent of \(S_i\).

A trace along an infinite branch \((S_i)_i\) is a sequence \((t_i)_{i \geq n}\) such that:

1. \(t_i\) is a precursor of \(t_{i+1}\); or
2. \(t_{i+1} < t_i\) occurs in the antecedent of \(S_i\). (a ‘progress point’)

Definition (\(\infty\)-proofs)
A \(\infty\)-proof (or just ‘proof’) is a preproof where each infinite branch has an infinitely progressing trace.
Irrationality of $\sqrt{2}$ again

\[
\begin{align*}
\vdots \\
\frac{b^2 = 2c^2}{c < a, 4c^2 = 2b^2} \\
\Rightarrow 2 \text{ is prime} & \quad \exists x < a. a = 2x, a^2 = 2b^2 \\
\frac{a^2 = 2b^2}{\Rightarrow \forall x, y. x^2 \neq 2y^2}
\end{align*}
\]
Irrationality of $\sqrt{2}$ again

\[
\begin{align*}
\therefore \quad b^2 &= 2c^2 \Rightarrow \\
\therefore \quad c < a, \ 4c^2 &= 2b^2 \Rightarrow \\
\Rightarrow \ 2 \text{ is prime} \quad \exists x < a. a = 2x, a^2 = 2b^2 \Rightarrow \\
\therefore \quad a^2 = 2b^2 \Rightarrow \\
\Rightarrow \quad \forall x, y. x^2 \neq 2y^2
\end{align*}
\]

There is an infinitely progressing trace $(a, c, b)^\omega$. 
Soundness of $\infty$-proofs

**Theorem (folklore)**

*If $A$ has a $\infty$-proof, then $\mathbb{N} \models A$.***
Soundness of $\infty$-proofs

**Theorem (folklore)**

*If* $A$ *has a $\infty$-proof, then* $\mathbb{N} \models A$.

**Proof idea.**

- Suppose otherwise, and build a branch of invalid sequents $(S_i)_i$.
- Simultaneously build assignments $\rho_i$ witnessing the invalidity.
Soundness of $\infty$-proofs

Theorem (folklore)
If $A$ has a $\infty$-proof, then $\mathbb{N} \models A$.

Proof idea.

- Suppose otherwise, and build a branch of invalid sequents $(S_i)_i$.
- Simultaneously build assignments $\rho_i$ witnessing the invalidity.
- By definition, there is an infinitely progressing trace $(t_i)_{i \geq n}$ along $(S_i)_i$.
- Can induce an infinite descending sequence $\rho_{i_1}(t_{i_1}) > \rho_{i_2}(t_{i_2}) > \cdots$
A finitary fragment: the cyclic proofs

**Definition**

A cyclic (or regular) proof is a $\infty$-proof with only finitely many distinct subtrees.

$CA$ is the theory of sentences that have cyclic proofs.

**Proposition (folklore)**

We can effectively check if a finite labelled graph is a correct cyclic proof.

**Proof.**

Let $\pi$ be a regular preproof. Define:

- $A_{\pi}^b$ (deterministic) Büchi automaton recognising infinite branches of $\pi$.
- $A_{\pi}^t$ NBA recognising branches of $\pi$ with an infinitely progressing trace.

Now simply check if $L(A_{\pi}^b) \subseteq L(A_{\pi}^t)$.

NB: inclusion of Büchi automata is $PSPACE$-complete.
Definition
A cyclic (or regular) proof is a $\infty$-proof with only finitely many distinct subtrees.
Definition
A cyclic (or regular) proof is a $\infty$-proof with only finitely many distinct subtrees. CA is the theory of sentences that have cyclic proofs.

Proposition (folklore)
We can effectively check if a finite labelled graph is a correct cyclic proof.
Definition
A cyclic (or regular) proof is a $\infty$-proof with only finitely many distinct subtrees. CA is the theory of sentences that have cyclic proofs.

Proposition (folklore)
We can effectively check if a finite labelled graph is a correct cyclic proof.

Proof.
Let $\pi$ be a regular preproof. Define:

- $A_b^\pi$ a (deterministic) Büchi automaton recognising infinite branches of $\pi$.
- $A_t^\pi$ a NBA recognising branches of $\pi$ with an infinitely progressing trace.

Now simply check if $L(A_b^\pi) \subseteq L(A_t^\pi)$. □
A finitary fragment: the cyclic proofs

Definition
A cyclic (or regular) proof is a $\infty$-proof with only finitely many distinct subtrees. CA is the theory of sentences that have cyclic proofs.

Proposition (folklore)
We can effectively check if a finite labelled graph is a correct cyclic proof.

Proof.
Let $\pi$ be a regular preproof. Define:
- $A_b^\pi$ a (deterministic) Büchi automaton recognising infinite branches of $\pi$.
- $A_t^\pi$ a NBA recognising branches of $\pi$ with an infinitely progressing trace.

Now simply check if $L(A_b^\pi) \subseteq L(A_t^\pi)$. □

NB: inclusion of Büchi automata is PSPACE-complete.
Outline

1. Peano and Cyclic Arithmetic
2. Summary of previous work and contributions
3. From induction to cycles
4. From cycles to induction
5. Some further results
6. Conclusions
Previous work

Theorem (Simpson '/one.lf/one.lf)

\[ \text{CA} = \text{PA} \]

- Formalises soundness argument for \( \infty \)-proofs in an appropriate fragment of SO arithmetic (\( \text{ACA} / \text{zero.lf} \)).
- Basic automaton theory for \( \omega \)-languages, can be carried out in \( \text{ACA} / \text{zero.lf} \).
- The result for \( \text{PA} \) is obtained by conservativity of \( \text{ACA} / \text{zero.lf} \) over \( \text{PA} \).
- Possibly non-elementary blowup in proof size, due to non-uniformity.

Theorem (Implicit in Berardi & Tatsuta '/one.lf/seven.lf)

\[ \text{CA} + I = \text{PA} + I \]

- 'Structural' argument, relying on proof-level manipulations.
- Relies on some nontrivial infinitary combinatorics specialised to arithmetic.
- High logical complexity.
Previous work

Theorem (Simpson ’11)
CA = PA.
Previous work

Theorem (Simpson ’11)
CA = PA.

- Formalises soundness argument for ∞-proofs in an appropriate fragment of SO arithmetic (ACA₀).
- (Basic automaton theory for ω-languages, can be carried out in ACA₀.)
Previous work

**Theorem (Simpson ’11)**

$\text{CA} = \text{PA}$.

- Formalises soundness argument for $\infty$-proofs in an appropriate fragment of SO arithmetic ($\text{ACA}_0$).
- (Basic *automaton theory* for $\omega$-languages, can be carried out in $\text{ACA}_0$.)
- The result for PA is obtained by *conservativity* of $\text{ACA}_0$ over PA.
Previous work

Theorem (Simpson ’11)

\[ \text{CA} = \text{PA}. \]

- Formalises soundness argument for \( \omega \)-proofs in an appropriate fragment of SO arithmetic (ACA\(_0\)).
- (Basic automaton theory for \( \omega \)-languages, can be carried out in ACA\(_0\).)
- The result for PA is obtained by conservativity of ACA\(_0\) over PA.
- Possibly non-elementary blowup in proof size, due to non-uniformity.
Previous work

Theorem (Simpson ’11)
CA = PA.

- Formalises soundness argument for $\infty$-proofs in an appropriate fragment of SO arithmetic (ACA₀).
- (Basic automaton theory for $\omega$-languages, can be carried out in ACA₀.)
- The result for PA is obtained by conservativity of ACA₀ over PA.
- Possibly non-elementary blowup in proof size, due to non-uniformity.

Theorem (Implicit in Berardi & Tatsuta ’17)
CA + $\mathcal{I}$ = PA + $\mathcal{I}$ for any set of Martin-Löf ordinary inductive definitions $\mathcal{I}$ and their associated rules.

- ‘Structural’ argument, relying on proof-level manipulations.
Previous work

Theorem (Simpson ’11)
CA = PA.

- Formalises soundness argument for \( \omega \)-proofs in an appropriate fragment of SO arithmetic (ACA\(_\omega\)).
- (Basic automaton theory for \( \omega \)-languages, can be carried out in ACA\(_\omega\).)
- The result for PA is obtained by conservativity of ACA\(_\omega\) over PA.
- Possibly non-elementary blowup in proof size, due to non-uniformity.

Theorem (Implicit in Berardi & Tatsuta ’17)
CA + \( \mathcal{I} \) = PA + \( \mathcal{I} \) for any set of Martin-Löf ordinary inductive definitions \( \mathcal{I} \) and their associated rules.

- ‘Structural’ argument, relying on proof-level manipulations.
- Relies on some nontrivial infinitary combinatorics specialised to arithmetic.
Previous work

Theorem (Simpson ’11)
\[ \text{CA} = \text{PA}. \]

- Formulates soundness argument for \( \infty \)-proofs in an appropriate fragment of SO arithmetic (ACA\(_0\)).
- (Basic automaton theory for \( \omega \)-languages, can be carried out in ACA\(_0\).)
- The result for PA is obtained by conservativity of ACA\(_0\) over PA.
- Possibly non-elementary blowup in proof size, due to non-uniformity.

Theorem (Implicit in Berardi & Tatsuta ’17)
\[ \text{CA} + \mathcal{I} = \text{PA} + \mathcal{I} \text{ for any set of Martin-Löf ordinary inductive definitions } \mathcal{I} \text{ and their associated rules.} \]

- ‘Structural’ argument, relying on proof-level manipulations.
- Relies on some nontrivial infinitary combinatorics specialised to arithmetic.
- High logical complexity.
Some questions

**Definition**
Write $C\Sigma_n$ for the theory axiomatised by the *universal closures* of CA proofs containing only $\Sigma_n$-formulae.

**NB:** A $C\Sigma_n$ proof of a $\Sigma_n$ sequent will contain only $\Sigma_n$ formulae anyway, by free-cut elimination.
Some questions

Definition
Write $C\Sigma_n$ for the theory axiomatised by the universal closures of CA proofs containing only $\Sigma_n$-formulae.

NB: A $C\Sigma_n$ proof of a $\Sigma_n$ sequent will contain only $\Sigma_n$ formulae anyway, by free-cut elimination.

Question (Simpson ‘17)

1. How does the logical complexity of CA and PA compare?
   Does $C\Sigma_m = I\Sigma_n$ for appropriately chosen $m, n$?
Some questions

Definition
Write $C\Sigma_n$ for the theory axiomatised by the universal closures of CA proofs containing only $\Sigma_n$-formulae.

NB: A $C\Sigma_n$ proof of a $\Sigma_n$ sequent will contain only $\Sigma_n$ formulae anyway, by free-cut elimination.

Question (Simpson ’17)

1. How does the logical complexity of CA and PA compare?
   Does $C\Sigma_m = I\Sigma_n$ for appropriately chosen $m, n$?

2. How does the proof complexity of PA and CA compare?
Some questions

Definition
Write $C\Sigma_n$ for the theory axiomatised by the **universal closures** of CA proofs containing only $\Sigma_n$-formulae.

**NB:** A $C\Sigma_n$ proof of a $\Sigma_n$ sequent will contain only $\Sigma_n$ formulae anyway, by **free-cut elimination**.

**Question (Simpson ’17)**

1. **How does the logical complexity of CA and PA compare?**
   Does $C\Sigma_m = I\Sigma_n$ for appropriately chosen $m, n$?

2. **How does the proof complexity of PA and CA compare?**

3. **Does cut-admissibility hold for any non-trivial fragment of CA?**
Digression: calibrating intuitions

It is tempting to think that

$$\sum_{n} = \sum_{n}.$$  

However this is not the case:

Example (Simpson /one.lf/seven.lf)

Recall the Ackermann-Péter function:

$$A(x, y) = 
\begin{cases} 
  y + 1 & \text{if } x = 0 \\
  A(x-1, A(x, y-1)) & \text{if } x > 0 \text{ and } y > 0 \\
  A(x-1, 0) & \text{if } x > 0 \text{ and } y = 0 
\end{cases}$$

Let $$A(x, y, z)$$ be an appropriate $$\Sigma/one.lf$$ formula computing its graph.

We have:
Digression: calibrating intuitions

It is tempting to think that $\Sigma_n = C\Sigma_n$. 

Example (Simpson '/one.lf/seven.lf)

Recall the Ackermann-Péter function:

$$A(x, y) = \begin{cases} y + 1, & x = 0 \\ A(x - 1, A(x, y - 1)), & x, y > 0 \end{cases}$$

Let $A(x, y, z)$ be an appropriate $\Sigma_n$ formula computing its graph.
It is tempting to think that $I \sum_n = C \sum_n$. However this is not the case:

**Example (Simpson ‘17)**

Recall the Ackermann-Péter function:

\[
A(x, y) = \begin{cases} 
  y + 1 & x = 0 \\
  A(x - 1, 1, z) & x > 0, y = 0 \\
  A(x - 1, A(x, y - 1)) & x, y > 0 
\end{cases}
\]

Let $A(x, y, z)$ be an appropriate $\Sigma_1$ formula computing its graph.
Digression: calibrating intuitions

It is tempting to think that $I\Sigma_n = C\Sigma_n$. However this is not the case:

**Example (Simpson ’17)**
Recall the Ackermann-Péter function:

$$A(x, y) = \begin{cases} 
    y + 1 & \text{if } x = 0 \\
    A(x - 1, 1, z) & \text{if } x > 0, y = 0 \\
    A(x - 1, A(x, y - 1)) & \text{if } x, y > 0 
\end{cases}$$

Let $A(x, y, z)$ be an appropriate $\Sigma_1$ formula computing its graph. We have:

\[
\begin{align*}
    x = 0 & \Rightarrow A(x, y, y+1) \\
    x > 0, y = 0 & \Rightarrow \exists z. A(x, y, z) \\
    x > 0, y > 0 & \Rightarrow \exists z. A(x, y, z) \\
    x > 0 & \Rightarrow \exists z. A(x, y, z) \\
    \Rightarrow \exists z. A(x, y, z)
\end{align*}
\]
On the other hand, some intuitions have simple proofs:

**Proposition**

For $n \geq 0$, $C\Sigma_n = C\Pi_n$. 
On the other hand, some intuitions have simple proofs:

**Proposition**

For \( n \geq 0 \), \( \Sigma_n = \Pi_n \).

**Proof.**

Simply replace every sequent \( \bar{p}, \Gamma \Rightarrow \Delta \) with \( \bar{p}, \bar{\Gamma} \Rightarrow \bar{\Delta} \), where \( \bar{p} \) exhausts all atomic formulae in the antecedent. \( \Box \)
Summary of contribution

Theorem

\[ \Sigma_n \subseteq \Pi_n \setminus \mathrm{one.lf} \]

\[ \supseteq : \text{by structural methods manipulating normal forms of inductive proofs.} \]

\[ \subseteq : \text{soundness argument can be formalised in conservative SO extensions.} \]

Theorem

PA and CA proof size differs only elementarily.

Proof idea.

Soundness argument can be made uniform in PA. Relies on:

• Deterministic acceptance of branch automaton is arithmetical.

• Well-foundedness of only finite ordinals is needed for the argument.

• \( \Rightarrow \) arithmetical approximation of non-deterministic acceptance.
Theorem

$C \Sigma_n = I \Sigma_{n+1}$, over $\Pi_{n+1}$ theorems.
Theorem
\[ C\Sigma_n = I\Sigma_{n+1}, \text{ over } \Pi_{n+1} \text{ theorems.} \]

\( \supseteq \): by structural methods manipulating normal forms of inductive proofs.
Summary of contribution

Theorem
\[ C\Sigma_n = I\Sigma_{n+1}, \text{ over } \Pi_{n+1} \text{ theorems.} \]

\supseteq: by structural methods manipulating normal forms of inductive proofs.
\subseteq: soundness argument can be formalised in conservative SO extensions.
Summary of contribution

**Theorem**

\[ C\Sigma_n = I\Sigma_{n+1}, \text{ over } \Pi_{n+1} \text{ theorems.} \]

\(\supseteq\): by structural methods manipulating normal forms of inductive proofs.

\(\subseteq\): soundness argument can be formalised in conservative SO extensions.

**Theorem**

PA and CA proof size differs only *elementarily.*
Summary of contribution

Theorem
$C\Sigma_n = I\Sigma_{n+1}$, over $\Pi_{n+1}$ theorems.

⊇: by structural methods manipulating normal forms of inductive proofs.
⊆: soundness argument can be formalised in conservative SO extensions.

Theorem
PA and CA proof size differs only elementarily.

Proof idea.
Soundness argument can be made uniform in PA. Relies on:
• Deterministic acceptance of branch automaton is arithmetical.
• Well-foundedness of only finite ordinals is needed for the argument.
• $\leadsto$ arithmetical approximation of non-deterministic acceptance.
1. Peano and Cyclic Arithmetic

2. Summary of previous work and contributions

3. From induction to cycles

4. From cycles to induction

5. Some further results

6. Conclusions
Main lemma

Let \( \pi \) be a \( \Pi_n \) proof, containing only \( \Pi_n \) formulae, of \( \Gamma, \forall x. A, \ldots, \forall x. A \), \( \Delta, \forall y. B, \ldots, \forall y. B \). Where \( \Gamma, \Delta, A_i, B_j \) are \( \Sigma_n \) and \( \vec{x}, \vec{y} \) occur only in \( \vec{A}, \vec{B} \) respectively.

Then there is a \( \Sigma_n \) derivation \( \lceil \pi \rceil \) of the form:

\[
\{ \Gamma \Rightarrow \Delta, A_i \}_{i \leq l} \quad \lceil \pi \rceil \quad \Gamma \Rightarrow \Delta, B_1, \ldots, B_m
\]

Moreover, no free variables of \((\text{one}.)\) occur as eigenvariables in \( \lceil \pi \rceil \).
Lemma
Let $\pi$ be a $\Pi_{n+1}$ proof, containing only $\Pi_{n+1}$ formulae, of

$$\Gamma, \forall x_1.A_1, \ldots, \forall x_l.A_l \Rightarrow \Delta, \forall y_1.B_1, \ldots, \forall y_m.B_m$$

(1)

where $\Gamma, \Delta, A_i, B_j$ are $\Sigma_n$ and $\vec{x}, \vec{y}$ occur only in $\vec{A}, \vec{B}$ respectively.
**Lemma**

Let $\pi$ be a $\Pi_{n+1}$ proof, containing only $\Pi_{n+1}$ formulae, of

$$\Gamma, \forall x_1.A_1, \ldots, \forall x_l.A_l \Rightarrow \Delta, \forall y_1.B_1, \ldots, \forall y_m.B_m$$

(1)

where $\Gamma, \Delta, A_i, B_j$ are $\Sigma_n$ and $\vec{x}, \vec{y}$ occur only in $\vec{A}, \vec{B}$ respectively.

Then there is a $C\Sigma_n$ derivation $[\pi]$ of the form:

Moreover, no free variables of (1) occur as eigenvariables in $[\pi]$. 

Diagram:

```
{Γ ⇒ Δ, Ai}_{i≤l}
\begin{array}{c}
\Gamma ⇒ Δ, B_1, \ldots, B_m
\end{array}
```

$[\pi]$
If \( \pi \) extends proofs \( \pi_0, \pi' \) by an induction step,

\[
\begin{align*}
\Gamma, \forall x.\bar{A} & \Rightarrow \Delta, \forall y.\bar{B}, \forall z.C(0) & \Gamma, \forall x.\bar{A}, \forall z.C(c) & \Rightarrow \Delta, \forall y.\bar{B}, \forall z.C(sc) \\
\Gamma, \forall x.\bar{A} & \Rightarrow \Delta, \forall y.\bar{B}, \forall x.C(t)
\end{align*}
\]

we define \( [\pi] \) to be the following cyclic proof:

\[
\begin{align*}
& \{\Gamma \Rightarrow \Delta, A_i\}_{i \leq l} \\
\Rightarrow & \quad [\pi_0] \\
= & \quad \begin{align*}
\Gamma & \Rightarrow \Delta, \bar{B}, A(0) \\
b & = 0, \Gamma \Rightarrow \Delta, \bar{B}, C(d) \\
\end{align*} \\
\text{sub} & \quad \Gamma \Rightarrow \Delta, \bar{B}, C(c) \\
\{\Gamma \Rightarrow \Delta, A_i\}_{i \leq l} & \quad [\pi'], \bar{B} \\
\text{sub} & \quad c < d, \Gamma \Rightarrow \Delta, \bar{B}, C(sc) \\
d & = sc, \Gamma \Rightarrow \Delta, \bar{B}, C(d) \quad \cdot
\end{align*}
\]
Outline

1. Peano and Cyclic Arithmetic
2. Summary of previous work and contributions
3. From induction to cycles
4. From cycles to induction
5. Some further results
6. Conclusions
Reverse mathematics of \( \omega \)-word automata

Reason about infinite words/sets in conservative SO extensions of FO arithmetic.

For an appropriate formalisation of NBA complementation, we have:

Theorem (Kolodziejczyk, Michalewski, Pradic & Skrzypczak)

\[ \text{RCA/zero.lf} + \Sigma/two.lf - \text{IND} \vdash \forall \text{NBA} A. \forall X. (X \in L(A^c) \equiv X \not\in L(A)) \]

Moreover, for each NBA \( A \), we have:

\[ \text{RCA/zero.lf} \vdash \forall X. (X \in L(A^c) \equiv X \not\in L(A)) \]

NB: \( /three.lf \) is implicit in that work. It is not trivial!
Reverse mathematics of $\omega$-word automata

Reason about infinite words/sets in conservative SO extensions of FO arithmetic.

\[
\text{RCA}_0 \cong I\Sigma_1 \cong \text{primitive recursive arithmetic}
\]
Reverse mathematics of $\omega$-word automata

Reason about infinite words/sets in \textit{conservative SO extensions} of FO arithmetic.

\[ \text{RCA}_0 \cong I\Sigma_1 \cong \text{primitive recursive arithmetic} \]

For an appropriate formalisation of \textit{NBA complementation}, we have:

\textbf{Theorem (Kolodziejczyk, Michalewski, Pradic & Skrzypczak ’16)}

\[ \text{RCA}_0 + \Sigma_2\text{-IND} \vdash \forall \text{NBA } \mathcal{A}. \forall X. (X \in \mathcal{L}(\mathcal{A}^c) \equiv X \notin \mathcal{L}(\mathcal{A})) \quad (2) \]
Reverse mathematics of $\omega$-word automata

Reason about infinite words/sets in conservative SO extensions of FO arithmetic.

\[ \text{RCA}_0 \cong I\Sigma_1 \cong \text{primitive recursive arithmetic} \]

For an appropriate formalisation of NBA complementation, we have:

**Theorem (Kolodziejczyk, Michalewski, Pradic & Skrzypczak ’16)**

\[ \text{RCA}_0 + \Sigma_2\text{-IND} \vdash \forall \text{NBA } A. \forall X. (X \in \mathcal{L}(A^c) \equiv X \notin \mathcal{L}(A)) \]

(2)

Moreover, for each NBA $A$, we have:

\[ \text{RCA}_0 \vdash \forall X. (X \in \mathcal{L}(A^c) \equiv X \notin \mathcal{L}(A)) \]

(3)
Reverse mathematics of $\omega$-word automata

Reason about infinite words/sets in conservative SO extensions of FO arithmetic.

$$\text{RCA}_0 \cong I\Sigma_1 \cong \text{primitive recursive arithmetic}$$

For an appropriate formalisation of NBA complementation, we have:

**Theorem (Kolodziejczyk, Michalewski, Pradic & Skrzypczak ’16)**

$$\text{RCA}_0 + \Sigma_2\text{-IND} \vdash \forall \text{ NBA } A. \forall X. (X \in L(A^c) \equiv X \notin L(A)) \quad (2)$$

Moreover, for each NBA $A$, we have:

$$\text{RCA}_0 \vdash \forall X. (X \in L(A^c) \equiv X \notin L(A)) \quad (3)$$

**NB:** (3) is implicit in that work. It is not trivial!
Write $\text{ArAcc}(X, \mathcal{A}/\text{two.lf})$ for:

"eventually, there are runs of $X$ on $\mathcal{A}/\text{two.lf}$ hitting final states arbitrarily often".

Theorem $I_{\Sigma/\text{one.lf} + \mathcal{A}/\text{two.lf}}$ "has a complement" proves:

$\forall \mathcal{DBA} \mathcal{A}/\text{one.lf}. (\mathcal{A}/\text{one.lf} \subseteq \mathcal{A}/\text{two.lf} \land X \in L(\mathcal{A}/\text{one.lf})) \supset \text{ArAcc}(X, \mathcal{A}/\text{two.lf})$

- $X \in L(\mathcal{A}/\text{one.lf})$ is arithmetical due to determinism.
- (Emptiness, unions and intersections of NBA formalisable in $\text{RCA}/\text{zero.lf}$.)

The soundness argument of $C_{\Sigma/n + \mathcal{A}/\text{one.lf}}$ constructs a $\Delta_{n+}/\text{one.lf}$-definable invalid branch, so:

Corollary $\text{PA}/\text{one.lf}$ elementarily simulates $\text{CA}/\text{two.lf}$.

$I_{\Sigma/n + \mathcal{A}/\text{one.lf}} \supset C_{\Sigma/n}$. 

$\mathcal{A}/\text{two.lf}$
From cycles to induction

Write \( \text{ArAcc}(X, A_2) \) for:

“eventually, there are runs of \( X \) on \( A_2 \) hitting final states arbitrarily often”
Write $\text{ArAcc}(X, A_2)$ for:

“eventually, there are runs of $X$ on $A_2$ hitting final states arbitrarily often”

**Theorem**

$I\Sigma_1(X) + \text{“$A_2$ has a complement”}$ proves:

$$\forall \text{DBA } A_1. (\text{“$A_1 \subseteq A_2$” } \land \ X \in \mathcal{L}(A_1)) \supset \text{ArAcc}(X, A_2)$$
Write $\text{ArAcc}(X, A_2)$ for:

“eventually, there are runs of $X$ on $A_2$ hitting final states arbitrarily often”

Theorem

$I \Sigma_1(X) + \text{“$A_2$ has a complement”}$ proves:

$$\forall \text{DBA } A_1. (\text{“$A_1 \subseteq A_2$”} \land X \in \mathcal{L}(A_1)) \supset \text{ArAcc}(X, A_2)$$

- $X \in \mathcal{L}(A_1)$ is arithmetical due to determinism.
- (Emptiness, unions and intersections of NBA formalisable in $\text{RCA}_0$.)
Write $\text{ArAcc}(X, A_2)$ for:

“eventually, there are runs of $X$ on $A_2$ hitting final states arbitrarily often”

**Theorem**

$I \Sigma_1(X) + \text{“$A_2$ has a complement”}$ proves:

$$\forall \text{DBA } A_1 . ( \text{“} A_1 \subseteq A_2 \text{”} \land X \in \mathcal{L}(A_1) ) \supset \text{ArAcc}(X, A_2)$$

- $X \in \mathcal{L}(A_1)$ is arithmetical due to determinism.
- (Emptiness, unions and intersections of NBA formalisable in $\text{RCA}_0$.)

The soundness argument of $C \Sigma_n$ constructs a $\Delta_{n+1}$-definable invalid branch,
From cycles to induction

Write $\text{ArAcc}(X, A_2)$ for:

“eventually, there are runs of $X$ on $A_2$ hitting final states arbitrarily often”

**Theorem**

$I\Sigma_1(X) + \text{“}A_2 \text{ has a complement}\text{”}$ proves:

$$\forall \text{dba } A_1. (\text{“}A_1 \subseteq A_2\text{”} \land X \in \mathcal{L}(A_1)) \supset \text{ArAcc}(X, A_2)$$

- $X \in \mathcal{L}(A_1)$ is arithmetical due to determinism.
- (Emptiness, unions and intersections of NBA formalisable in RCA$_0$.)

The soundness argument of $C\Sigma_n$ constructs a $\Delta_{n+1}$-definable invalid branch, so:

**Corollary**

1. PA elementarily simulates CA.
2. $I\Sigma_{n+1} \supseteq C\Sigma_n$. 
Outline

1. Peano and Cyclic Arithmetic
2. Summary of previous work and contributions
3. From induction to cycles
4. From cycles to induction
5. Some further results
6. Conclusions
Computational aspects of CA

**Provably recursive functions of** $C\Delta_0$

- For $n \geq 1$, the provably recursive functions of $C\Sigma_n$ are just those of $I\Sigma_{n+1}$. 
Computational aspects of CA

**Provably recursive functions of $C\Delta_0$**

- For $n \geq 1$, the provably recursive functions of $C\Sigma_n$ are just those of $I\Sigma_{n+1}$.
- However $C\Delta_0$ is $\Pi_1$-axiomatised, so by Parikh’s theorem we have:

**Corollary**

The provably recursive functions of $C\Delta_0$ are just those of $I\Delta_0$, i.e. the linear-time hierarchy.
Computational aspects of CA

**Provably recursive functions of $C\Delta_0$**

- For $n \geq 1$, the provably recursive functions of $C\Sigma_n$ are just those of $I\Sigma_{n+1}$.
- However $C\Delta_0$ is $\Pi_1$-axiomatised, so by Parikh’s theorem we have:

**Corollary**

*The provably recursive functions of $C\Delta_0$ are just those of $I\Delta_0$, i.e. the linear-time hierarchy.*

**Failure of cut-admissibility**
Computational aspects of CA

**Provably recursive functions of $C\Delta_0$**

- For $n \geq 1$, the provably recursive functions of $C\Sigma_n$ are just those of $I\Sigma_{n+1}$.
- However $C\Delta_0$ is $\Pi_1$-axiomatised, so by Parikh’s theorem we have:

**Corollary**

The provably recursive functions of $C\Delta_0$ are just those of $I\Delta_0$, i.e. the linear-time hierarchy.

**Failure of cut-admissibility**

**Corollary**

For $n \geq 1$, the class of CA proofs with only $\Sigma_{n-1}$ cuts is not complete for $C\Sigma_n$. 

Computational aspects of CA

Provably recursive functions of $C\Delta_0$

- For $n \geq 1$, the provably recursive functions of $C\Sigma_n$ are just those of $I\Sigma_{n+1}$.
- However $C\Delta_0$ is $\Pi_1$-axiomatised, so by Parikh’s theorem we have:

Corollary

The provably recursive functions of $C\Delta_0$ are just those of $I\Delta_0$, i.e. the linear-time hierarchy.

Failure of cut-admissibility

Corollary

For $n \geq 1$, the class of CA proofs with only $\Sigma_{n-1}$ cuts is not complete for $C\Sigma_n$.

Proof.

- $I\Sigma_{n+1} \vdash \text{Con}_{I\Sigma_n}$ so $C\Sigma_n \vdash \text{Con}_{I\Sigma_n}$ by $\Pi_{n+1}$-conservativity.
Computational aspects of CA

**Provably recursive functions of $C\Delta_0$**

- For $n \geq 1$, the provably recursive functions of $C\Sigma_n$ are just those of $I\Sigma_{n+1}$.
- However $C\Delta_0$ is $\Pi_1$-axiomatised, so by Parikh’s theorem we have:

**Corollary**

The provably recursive functions of $C\Delta_0$ are just those of $I\Delta_0$, i.e. the linear-time hierarchy.

**Failure of cut-admissibility**

**Corollary**

For $n \geq 1$, the class of CA proofs with only $\Sigma_{n-1}$ cuts is not complete for $C\Sigma_n$.

**Proof.**

- $I\Sigma_{n+1} \vdash \text{Con}_{I\Sigma_n}$ so $C\Sigma_n \vdash \text{Con}_{I\Sigma_n}$ by $\Pi_{n+1}$-conservativity.
- On the other hand, $C\Sigma_{n-1} \not\vdash \text{Con}_{I\Sigma_n}$ since otherwise $I\Sigma_n \vdash \text{Con}_{I\Sigma_n}$.  

□
Metalogical aspects of CA

Reflection and consistency

**Corollary**

For \( n \geq 0 \), \( I \Sigma n + 1 \vdash \Pi n - Rfn \Sigma n \).

In particular we have \( I \Sigma n + 1 \vdash \text{Con} \Sigma n \).

**Proof.**

Otherwise \( C \Sigma n \vdash \text{Con} \Sigma n \) by \( \Pi n + 1 \)-conservativity.

Unsurprisingly, we have Gödel incompleteness for all fragments \( C \Sigma n \).

In particular, we have:

**Corollary**

For \( n \geq 0 \), \( I \Sigma n + 1 \nvdash \text{Con} \Sigma n \).
Reflection and consistency

Rephrasing our results in terms of logical strength, we have:

**Corollary**

For $n \geq 0$, $I\Sigma_{n+2} \vdash \Pi_{n+1}^{n+1}-\text{Rfn}_{\Sigma_n}$. 

Proof.

Otherwise $C\Sigma_n \vdash \text{Con}_C\Sigma_n$ by $\Pi_{n+1}^{n+1}$-conservativity.
**Reflection and consistency**

Rephrasing our results in terms of logical strength, we have:

**Corollary**

For \( n \geq 0 \), \( \text{I} \Sigma_{n+2} \vdash \Pi_{n+1} \text{-Rfn}_{C \Sigma_n} \). In particular we have \( \text{I} \Sigma_{n+2} \vdash \text{Con}_{C \Sigma_n} \).

**Incompleteness**
Reflection and consistency
Rephrasing our results in terms of logical strength, we have:

**Corollary**
For \( n \geq 0 \), \( I\Sigma_{n+2} \vdash \Pi_{n+1}\text{-Rfn}_{C\Sigma_n} \). In particular we have \( I\Sigma_{n+2} \vdash \text{Con}_{C\Sigma_n} \).

Incompleteness
Unsurprisingly, we have Gödel incompleteness for all fragments \( C\Sigma_n \).
Reflection and consistency

Rephrasing our results in terms of logical strength, we have:

Corollary

For $n \geq 0$, $I\Sigma_{n+2} \vdash \Pi_{n+1}$-Rfn$_{C\Sigma_n}$. In particular we have $I\Sigma_{n+2} \vdash \text{Con}_{C\Sigma_n}$.

Incompleteness

Unsurprisingly, we have Gödel incompleteness for all fragments $C\Sigma_n$.

In particular, we have:

Corollary

For $n \geq 0$, $I\Sigma_{n+1} \nvdash \text{Con}_{C\Sigma_n}$.
Reflection and consistency

Rephrasing our results in terms of logical strength, we have:

Corollary

For \( n \geq 0 \), \( \mathsf{IΣ}_{n+2} \vdash \Pi_{n+1} \text{-Rfn}_{\mathsf{CΣ}_n} \). In particular we have \( \mathsf{IΣ}_{n+2} \vdash \text{Con}_{\mathsf{CΣ}_n} \).

Incompleteness

Unsurprisingly, we have Gödel incompleteness for all fragments \( \mathsf{CΣ}_n \).

In particular, we have:

Corollary

For \( n \geq 0 \), \( \mathsf{IΣ}_{n+1} \nvdash \text{Con}_{\mathsf{CΣ}_n} \).

Proof.

Otherwise \( \mathsf{CΣ}_n \vdash \text{Con}_{\mathsf{CΣ}_n} \) by \( \Pi_{n+1} \)-conservativity.
Reverse mathematics of McNaughton’s theorem

In fact, there is a curious consequence for \( \omega \)-automaton theory.

Theorem
A natural formulation of McNaughton’s theorem, that every NBA has an equivalent deterministic parity automaton, is not provable in \( \text{RCA}_{\text{zero}} \).

Proof idea.
• If \( A_{\text{one}} \) is a DBA, we can check \( L(A_{\text{one}}) \subseteq L(A_{\text{two}}) \) by complementing \( A_{\text{one}} \) in \( \text{RCA}_{\text{zero}} \) and checking for universality of \( A_{\text{one}} \cup A_{\text{two}} \).
• (Given McNaughton, we may check universality already in \( \text{RCA}_{\text{zero}} \)).
• This allows us to formalise, say, the soundness of \( C_{\Delta} \) already in \( \text{I}_{\Sigma} \), contradicting Gödel’s second incompleteness result for \( C_{\Delta} \).

This was not known before!
Reverse mathematics of McNaughton’s theorem

In fact, there is a curious consequence for $\omega$-automaton theory.
In fact, there is a curious consequence for $\omega$-automaton theory.

**Theorem**

A natural formulation of McNaughton’s theorem, that every NBA has an equivalent deterministic parity automaton, is *not provable in RCA$_0$*. 
In fact, there is a curious consequence for ω-automaton theory.

**Theorem**

A natural formulation of McNaughton’s theorem, that every NBA has an equivalent deterministic parity automaton, is not provable in RCA₀.

**Proof idea.**

- If $A_1$ is a DBA, we can check $\mathcal{L}(A_1) \subseteq \mathcal{L}(A_2)$ by complementing $A_1$ in RCA₀ and checking for universality of $A_1^c \cup A_2$. 
In fact, there is a curious consequence for ω-automaton theory.

**Theorem**

A natural formulation of McNaughton’s theorem, that every NBA has an equivalent deterministic parity automaton, is not provable in RCA₀.

**Proof idea.**

- If $A_1$ is a DBA, we can check $\mathcal{L}(A_1) \subseteq \mathcal{L}(A_2)$ by complementing $A_1$ in RCA₀ and checking for universality of $A_1^c \cup A_2$.
- (Given McNaughton, we may check universality already in RCA₀).
Reverse mathematics of McNaughton’s theorem

In fact, there is a curious consequence for $\omega$-automaton theory.

**Theorem**

A natural formulation of McNaughton’s theorem, that every NBA has an equivalent deterministic parity automaton, is **not provable in $\text{RCA}_0$**.

**Proof idea.**

- If $A_1$ is a DBA, we can check $\mathcal{L}(A_1) \subseteq \mathcal{L}(A_2)$ by complementing $A_1$ in $\text{RCA}_0$ and checking for universality of $A_1^c \cup A_2$.
- (Given McNaughton, we may check universality already in $\text{RCA}_0$).
- This allows us to formalise, say, the soundness of $C\Delta_0$ already in $I\Sigma_1$, contradicting Gödel’s second incompleteness result for $C\Delta_0$.

□
In fact, there is a curious consequence for \(\omega\)-automaton theory.

**Theorem**

A natural formulation of McNaughton’s theorem, that every NBA has an equivalent deterministic parity automaton, is *not provable in RCA\(_0\)*.

**Proof idea.**

- If \(A_1\) is a DBA, we can check \(L(A_1) \subseteq L(A_2)\) by complementing \(A_1\) in RCA\(_0\) and checking for universality of \(A_1^c \cup A_2\).
- (Given McNaughton, we may check universality already in RCA\(_0\)).
- This allows us to formalise, say, the soundness of \(C\Delta_0\) already in \(I\Sigma_1\), contradicting Gödel’s second incompleteness result for \(C\Delta_0\).

This was *not known* before!
Outline

1. Peano and Cyclic Arithmetic
2. Summary of previous work and contributions
3. From induction to cycles
4. From cycles to induction
5. Some further results
6. Conclusions
Further directions - computational interpretations of proofs

What about cyclic versions of Gödel's System T?

⇝

Interestingly, Ackermann-Péter has a 'type-/zero.lf' cyclic proof:

Question

Does 'cyclic-T' exhibit a /one.lf-level improvement over T?

Work-in-progress:

a Dialectica-style functional interpretation of CA

/two.lf/eight.lf / /two.lf/nine.lf
Further directions - computational interpretations of proofs

What about cyclic versions of Gödel’s System $T$?
What about cyclic versions of Gödel’s System T?
⇝ recent progress with the Lyonese school.
What about cyclic versions of Gödel’s System T?
\[ \leadsto \text{recent progress with the Lyonese school.} \]

Interestingly, Ackermann-Péter has a ‘type-0’ cyclic proof:

\[
\begin{align*}
\rightarrow 1 & \quad \rightarrow 1^* \\
\rightarrow 1 & \quad \rightarrow 1 \\
\ast_l & \quad 1^*, 1^* \rightarrow 1^* \\
\ast_l & \quad 1^* \rightarrow 1^* \\
w & \quad 1^*, 1^* \rightarrow 1^* \\
\end{align*}
\]

\[
\begin{align*}
\rightarrow 1 & \quad \rightarrow 1^* \\
\rightarrow 1^* & \quad 1^*, 1^* \rightarrow 1^* \\
\ast_l & \quad 1^*, 1^* \rightarrow 1^* \\
c & \quad 1^*, 1^* \rightarrow 1^* \\
w & \quad 1^*, 1^*, 1^* \rightarrow 1^* \\
c & \quad 1^*, 1^*, 1^* \rightarrow 1^* \\
\end{align*}
\]
Further directions - computational interpretations of proofs

What about cyclic versions of Gödel’s System $T$? \(\rightsquigarrow\) recent progress with the Lyonese school.

Interestingly, Ackermann-Péter has a ‘type-0’ cyclic proof:

\[
\begin{align*}
\rightarrow 1 & \rightarrow 1* \rightarrow 1 & (1) \\
\rightarrow 1* & \rightarrow 1* \rightarrow 1* & (2) \\
1* \rightarrow 1* & 1* \rightarrow 1* \rightarrow 1* & (3)
\end{align*}
\]

Question

Does ‘cyclic-$T$’ exhibit a 1-level improvement over $T$?
Further directions - computational interpretations of proofs

What about cyclic versions of Gödel’s System T?
⇒ recent progress with the Lyonese school.

Interestingly, Ackermann-Péter has a ‘type-O’ cyclic proof:

\[
\begin{array}{llllll}
\rightarrow & 1 & \rightarrow & 1^* & 1 & \rightarrow 1^* \\
\ast l & \rightarrow & 1^* & 1,1^* & \rightarrow & 1^* \\
\ast l & \rightarrow & 1^* & 1^*,1^* & \rightarrow & 1^* \\
\ast l & \rightarrow & 1^* & 1^*,1^* & \rightarrow & 1^* \\
\end{array}
\]

Question

Does ‘cyclic-T’ exhibit a 1-level improvement over T?

Work-in-progress: a Dialectica-style functional interpretation of CA.
Corollary \( \Sigma^n \) is precisely the \( \Pi^n + \) one.lf consequences of \( \Sigma^n + \) one.lf.

Proof complexity differs only elementarily. In fact:

Corollary \( \text{PA} \) exponentially simulates \( \text{CA} \). This is optimal, unless there is a more efficient way to check cyclic proof soundness.

Question

What is the logical strength of McNaughton's theorem, in general?

Thank you.
Summary and open questions

Optimal logical complexity result. In fact:

Corollary

\(C \Sigma_n\) is precisely the \(\Pi_{n+1}\) consequences of \(I \Sigma_{n+1}\).
Summary and open questions

Optimal logical complexity result. In fact:

**Corollary**

$C \Sigma_n$ is precisely the $\Pi_{n+1}$ consequences of $I \Sigma_{n+1}$.

Proof complexity differs only elementarily. In fact:

**Corollary**

PA *exponentially simulates* CA. *This is optimal, unless there is a more efficient way to check cyclic proof soundness.*
Summary and open questions

Optimal logical complexity result. In fact:

**Corollary**

$CΣ_n$ is precisely the $Π_{n+1}$ consequences of $IΣ_{n+1}$.

Proof complexity differs only elementarily. In fact:

**Corollary**

$PA$ *exponentially simulates* $CA$. This is optimal, unless there is a more efficient way to check cyclic proof soundness.

**Question**

What is the *logical strength* of McNaughton’s theorem, in general?
Summary and open questions

Optimal logical complexity result. In fact:

Corollary

$C\Sigma_n$ is precisely the $\Pi_{n+1}$ consequences of $I\Sigma_{n+1}$.

Proof complexity differs only elementarily. In fact:

Corollary

PA \textit{exponentially simulates} CA. This is optimal, unless there is a more efficient way to check cyclic proof soundness.

Question

What is the logical strength of McNaughton’s theorem, in general?

Thank you.