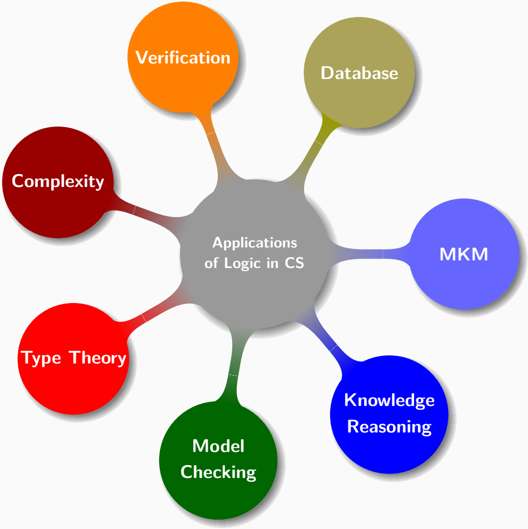
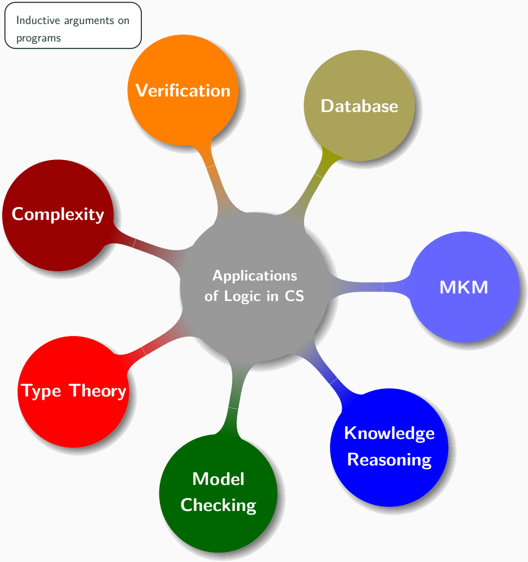
Induction, Transitive Closure and Cycles

Liron Cohen, Cornell University,
Reuben Rowe, University of Kent

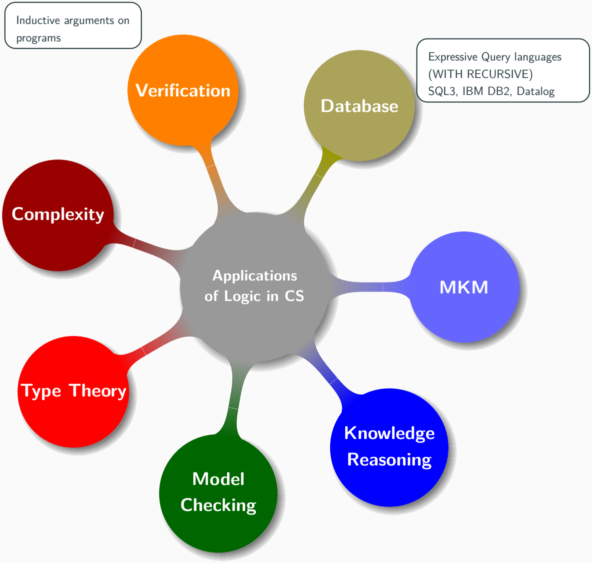
ASL North American Annual Meeting, 2018



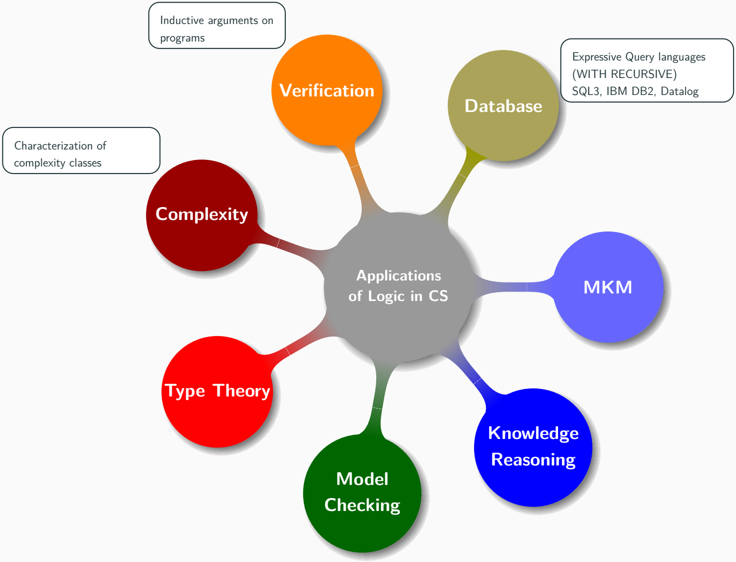
Halpern, Harper, Immerman, Kolaitis, Vardi, Vianu. On the unusual effectiveness of logic in computer science, 2001



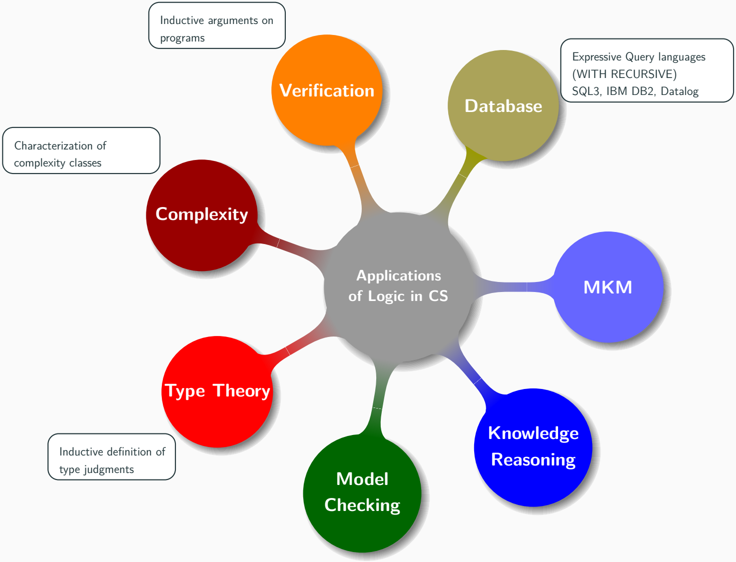
Halpern, Harper, Immerman, Kolaitis, Vardi, Vianu. On the unusual effectiveness of logic in computer science, 2001



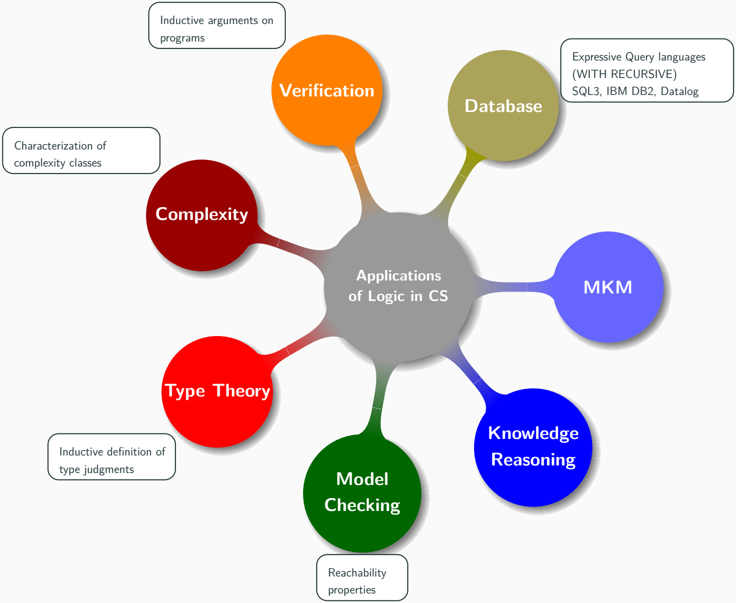
Halpern, Harper, Immerman, Kolaitis, Vardi, Vianu. On the unusual effectiveness of logic in computer science, 2001



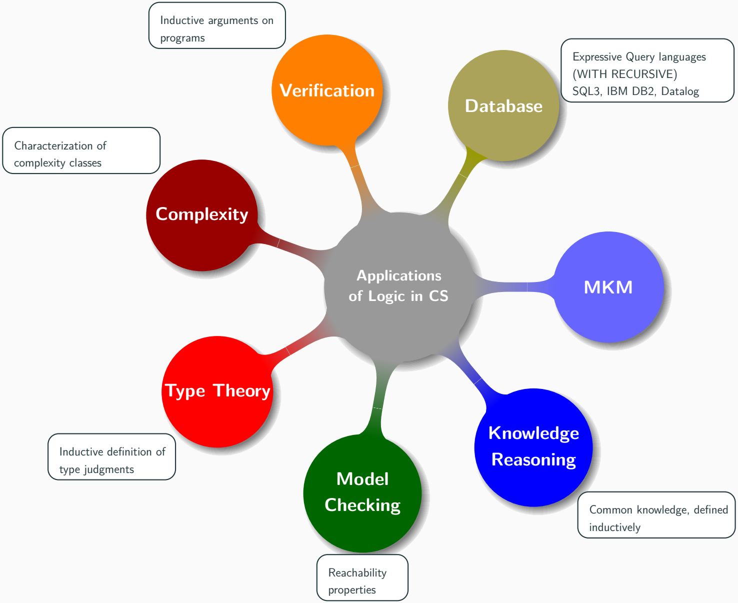
Halpern, Harper, Immerman, Kolaitis, Vardi, Vianu. On the unusual effectiveness of logic in computer science, 2001



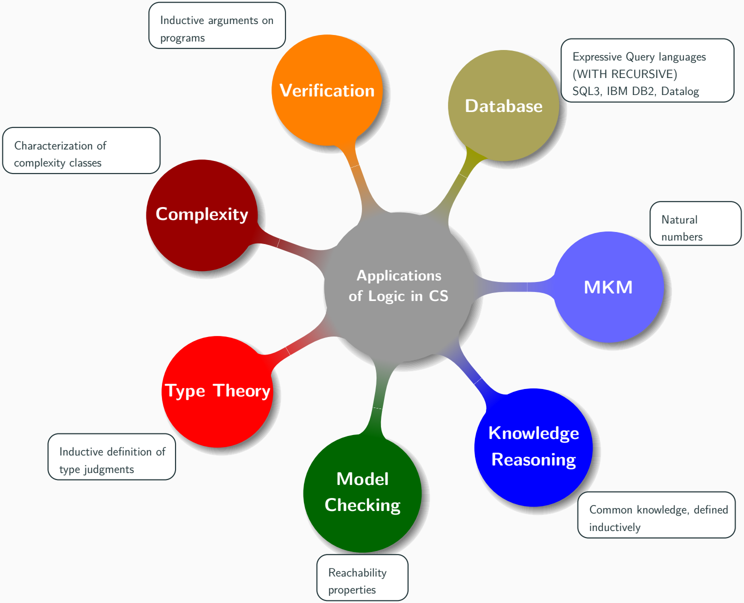
Halpern, Harper, Immerman, Kolaitis, Vardi, Vianu. On the unusual effectiveness of logic in computer science, 2001



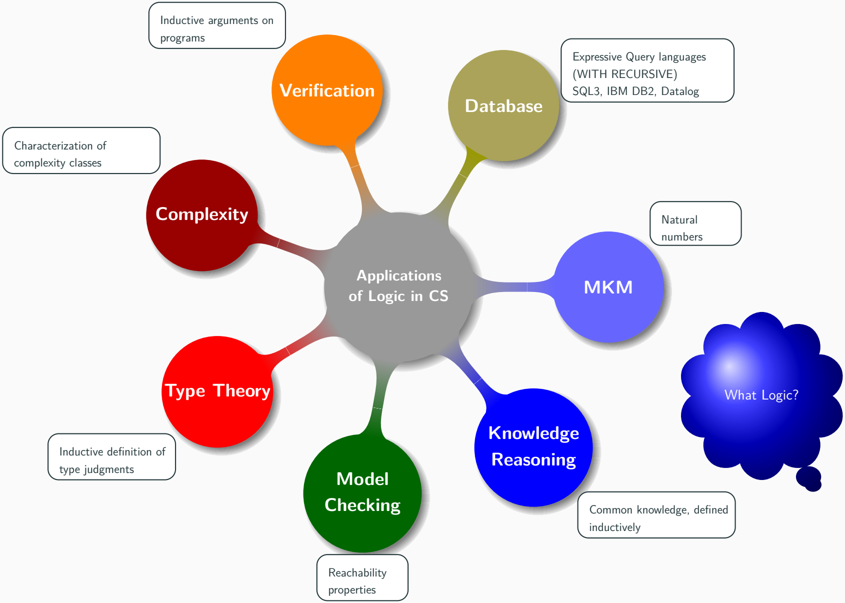
Halpern, Harper, Immerman, Kolaitis, Vardi, Vianu. On the unusual effectiveness of logic in computer science, 2001



Halpern, Harper, Immerman, Kolaitis, Vardi, Vianu. On the unusual effectiveness of logic in computer science, 2001



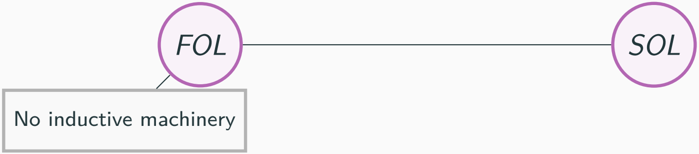
Halpern, Harper, Immerman, Kolaitis, Vardi, Vianu. On the unusual effectiveness of logic in computer science, 2001



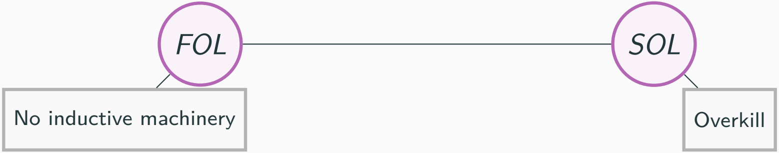
Halpern, Harper, Immerman, Kolaitis, Vardi, Vianu. On the unusual effectiveness of logic in computer science, 2001

What Logic?

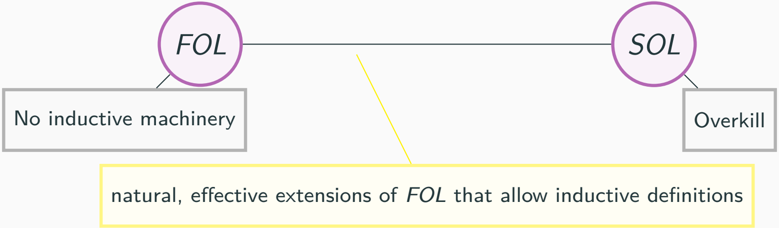
What Logic?



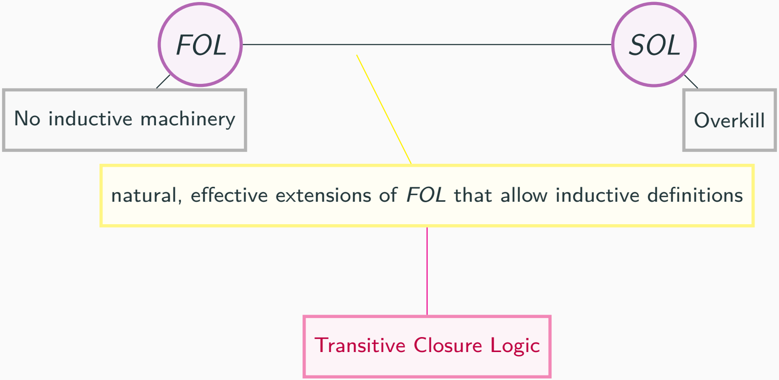
What Logic?



What Logic?


FOL
SOL
No inductive machinery
Overkill
natural, effective extensions of *FOL* that allow inductive definitions

What Logic?


FOL
SOL
No inductive machinery
Overkill
natural, effective extensions of *FOL* that allow inductive definitions
Transitive Closure Logic

Transitive Closure Logic

Transitive Closure Logic $= FOL +$ a transitive closure operator.

Transitive Closure Logic

Transitive Closure Logic $= FOL +$ a transitive closure operator.

The transitive closure R^* of binary relation R is defined by:

$$R^* = \bigcup R^{(n)}$$

where $R^{(0)} = Id, R^{(n+1)} = R^{(n)} \circ R$.

Transitive Closure Logic

Transitive Closure Logic $= FOL +$ a transitive closure operator.

The transitive closure R^* of binary relation R is defined by:

$$R^* = \bigcup R^{(n)}$$

where $R^{(0)} = Id, R^{(n+1)} = R^{(n)} \circ R$.

Alternatively,

$$R^* = Id \cup \bigcap \{S \mid R \cup S \circ R \subseteq S\}$$

(Least fixed point of the composition operator)

Why Transitive Closure Logic?

- The concept of the transitive closure is truly basic.

Why Transitive Closure Logic?

- The concept of the transitive closure is truly basic.
  - Being a 'descendent of'
  - The natural numbers
  - Well-formed formulas

Why Transitive Closure Logic?

- The concept of the transitive closure is truly basic.
  - Being a ‘descendent of’
  - The natural numbers
  - Well-formed formulas
- A minimal extension.

Why Transitive Closure Logic?

- The concept of the transitive closure is truly basic.
  - Being a ‘descendent of’
  - The natural numbers
  - Well-formed formulas
- A minimal extension.
  - A special case of a least fixed point.

Why Transitive Closure Logic?

- The concept of the transitive closure is truly basic.
  - Being a 'descendent of'
  - The natural numbers
  - Well-formed formulas
- A minimal extension.
  - A special case of a least fixed point.
- Equivalent to other extensions of *FOL*, but the most convenient from a proof theoretical perspective.

Why Transitive Closure Logic?

- The concept of the transitive closure is truly basic.
    - Being a 'descendent of'
    - The natural numbers
    - Well-formed formulas
- A minimal extension.
    - A special case of a least fixed point.
- Equivalent to other extensions of *FOL*, but the most convenient from a proof theoretical perspective.
- Captures inductive principles in a uniform way.

Why Transitive Closure Logic?

- The concept of the transitive closure is truly basic.
  - Being a ‘descendent of’
  - The natural numbers
  - Well-formed formulas
- A minimal extension.
  - A special case of a least fixed point.
- Equivalent to other extensions of *FOL*, but the most convenient from a proof theoretical perspective.
- Captures inductive principles in a uniform way.
  - Not parametrized by a set of inductive principles.

The Language

The Language

The language $\mathcal{L}_{TC}$ is defined as $\mathcal{L}_{FOL}$, with the additional clause:

- $(RTC_{x,y}\varphi)(s,t)$ is a formula,
  for φ a formula, x, y distinct variables, and s, t terms.
  (x, y become bound in this formula.)

The Language

The language $\mathcal{L}_{TC}$ is defined as $\mathcal{L}_{FOL}$, with the additional clause:

- $(RTC_{x,y}\varphi)(s,t)$ is a formula,
  for φ a formula, x, y distinct variables, and s, t terms.
  (x, y become bound in this formula.)

Allows for:

- Rich testing
- Nested *RTC*

The Semantics

The Intended Meaning of $(RTC_{x,y}\varphi)(s,t)$

$$s = t \lor \varphi(s,t) \lor \exists w_1.\varphi(s,w_1) \land \varphi(w_1,t) \\ \lor \exists w_1 \exists w_2.\varphi(s,w_1) \land \varphi(w_1,w_2) \land \varphi(w_2,t) \lor \ldots$$

The Semantics

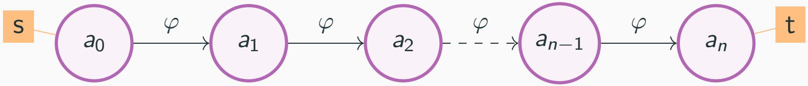
The Intended Meaning of $(RTC_{x,y}\varphi)(s,t)$

$$s = t \lor \varphi(s,t) \lor \exists w_1.\varphi(s,w_1) \land \varphi(w_1,t)$$
$$\lor \exists w_1 \exists w_2.\varphi(s,w_1) \land \varphi(w_1,w_2) \land \varphi(w_2,t) \lor \ldots$$

Formal Definition

Let M be a structure for $\mathcal{L}_{TC}$ and v an assignment in M.

$M, v \models (RTC_{x,y}\varphi)(s,t)$ iff there exist $a_0, \ldots a_n \in D$ s.t.
$v[s] = a_0$; $v[t] = a_n$; $M, v[x := a_i, y := a_{i+1}] \models \varphi$ for $0 \leq i < n$.



The Semantics

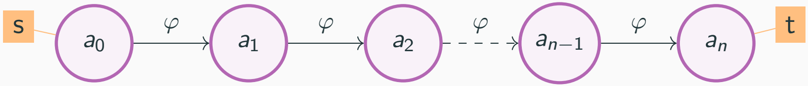
The Intended Meaning of $(RTC_{x,y}\varphi)(s,t)$

$$s = t \vee \varphi(s,t) \vee \exists w_1.\varphi(s,w_1) \wedge \varphi(w_1,t)$$
$$\vee \exists w_1 \exists w_2.\varphi(s,w_1) \wedge \varphi(w_1,w_2) \wedge \varphi(w_2,t) \vee \dots$$

Formal Definition

Let M be a structure for $\mathcal{L}_{TC}$ and v an assignment in M.

$M, v \models (RTC_{x,y}\varphi)(s,t)$ iff there exist $a_0, \dots a_n \in D$ s.t. $v[s] = a_0$; $v[t] = a_n$; $M, v[x := a_i, y := a_{i+1}] \models \varphi$ for $0 \leq i < n$.



$M, v \models (RTC_{x,y}\varphi)(s,t)$ provided for every $A \subseteq D$, if $v(s) \in A$ and $\forall a, b \in D: (a \in A \wedge M, v[x := a, y := b] \models \varphi) \rightarrow b \in A$, then $v(t) \in A$.

Expressive Power

- The reflexive and the non-reflexive TC operators are equivalent (assuming equality).

Expressive Power

- The reflexive and the non-reflexive TC operators are equivalent (assuming equality).

Theorem [Avron, '03]

All recursive functions and relations are definable in $\mathcal{L}_{TC}^{\{0,s\}}$

Expressive Power

- The reflexive and the non-reflexive TC operators are equivalent (assuming equality).

Theorem [Avron, '03]

All recursive functions and relations are definable in $\mathcal{L}_{TC}^{\{0,s\}}$ (with pairs)

Expressive Power

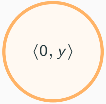
- The reflexive and the non-reflexive TC operators are equivalent (assuming equality).

Theorem [Avron, '03]

All recursive functions and relations are definable in $\mathcal{L}_{TC}^{\{0,s\}}$ (with pairs)

- $+$ is definable in $\mathcal{L}_{TC}^{\{0,s\}}$ (with pairs) by:

$$x = y + z \iff (RTC_{u,v}v.1 = s(u.1) \wedge v.2 = s(u.2))((0, y), (z, x))$$



Expressive Power

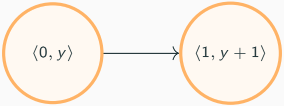
- The reflexive and the non-reflexive TC operators are equivalent (assuming equality).

Theorem [Avron, '03]

All recursive functions and relations are definable in $\mathcal{L}_{TC}^{\{0,s\}}$ (with pairs)

- $+$ is definable in $\mathcal{L}_{TC}^{\{0,s\}}$ (with pairs) by:

$$x = y + z \Longleftrightarrow (RTC_{u,v} v.1 = s(u.1) \wedge v.2 = s(u.2))((0, y), (z, x))$$



Expressive Power

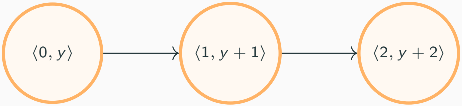
- The reflexive and the non-reflexive TC operators are equivalent (assuming equality).

Theorem [Avron, '03]

All recursive functions and relations are definable in $\mathcal{L}_{TC}^{\{0,s\}}$ (with pairs)

- $+$ is definable in $\mathcal{L}_{TC}^{\{0,s\}}$ (with pairs) by:

$$x = y + z \iff (RTC_{u,v} v.1 = s(u.1) \wedge v.2 = s(u.2))((0,y),(z,x))$$



Expressive Power

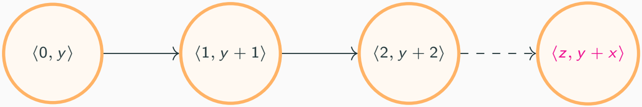
- The reflexive and the non-reflexive TC operators are equivalent (assuming equality).

Theorem [Avron, '03]

All recursive functions and relations are definable in $\mathcal{L}_{TC}^{\{0,s\}}$ (with pairs)

- $+$ is definable in $\mathcal{L}_{TC}^{\{0,s\}}$ (with pairs) by:

$$x = y + z \iff (RTC_{u,v} v.1 = s(u.1) \wedge v.2 = s(u.2))((0, y), (z, x))$$



Expressive Power

Categorical Characterization of the Natural Numbers

$$\forall x\,(s\,(x) \neq 0)$$

$$\forall x \forall y\,(s\,(x) = s\,(y) \rightarrow x = y)$$

$$\forall x\,(RTC_{w,u}\,(s(w) = u))\,(0, x)$$

Expressive Power

Categorical Characterization of the Natural Numbers

$$\forall x\, (s\,(x) \neq 0)$$
$$\forall x \forall y\, (s\,(x) = s\,(y) \to x = y)$$
$$\forall x\, (RTC_{w,u}\, (s(w) = u))\, (0, x)$$

Corollaries:

- The upward Löwenheim-Skolem theorem fails for TC-logic.

Expressive Power

Categorical Characterization of the Natural Numbers

$$\forall x\,(s\,(x) \neq 0)$$

$$\forall x \forall y\,(s\,(x) = s\,(y) \rightarrow x = y)$$

$$\forall x\,(RTC_{w,u}\,(s(w) = u))\,(0, x)$$

Corollaries:

- The upward Löwenheim-Skolem theorem fails for TC-logic.
- TC-logic is not compact.

Expressive Power

Categorical Characterization of the Natural Numbers

$$\forall x \, (s\,(x) \neq 0)$$

$$\forall x \forall y \, (s\,(x) = s\,(y) \rightarrow x = y)$$

$$\forall x \, (RTC_{w,u}\,(s(w) = u))\,(0, x)$$

Corollaries:

- The upward Löwenheim-Skolem theorem fails for TC-logic.
- TC-logic is not compact.
- TC-logic is inherently incomplete.

Expressive Power

Categorical Characterization of the Natural Numbers

$$\forall x\,(s\,(x) \neq 0)$$

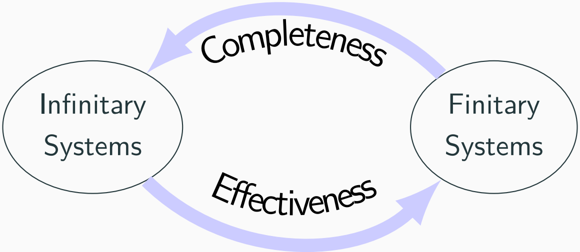
$$\forall x \forall y\,(s\,(x) = s\,(y) \rightarrow x = y)$$

$$\forall x\,(RTC_{w,u}\,(s(w) = u))\,(0, x)$$

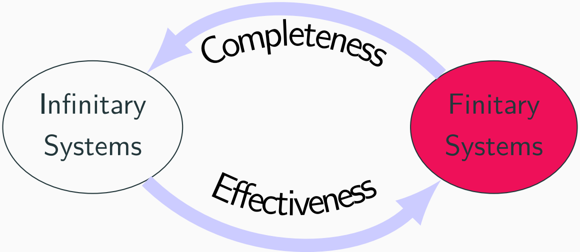
Corollaries:

- The upward Löwenheim-Skolem theorem fails for TC-logic.
- TC-logic is not compact.
- TC-logic is inherently incomplete.

Proof Theory



Proof Theory



The System $\mathcal{LK}_=$ [Gentzen, '34]

$$\frac{\psi, \Gamma \Rightarrow \Delta}{\varphi \wedge \psi, \Gamma \Rightarrow \Delta}\ (\wedge L_1) \qquad \frac{\varphi, \Gamma \Rightarrow \Delta}{\varphi \wedge \psi, \Gamma \Rightarrow \Delta}\ (\wedge L_2) \qquad \frac{\Gamma \Rightarrow \Delta, \varphi \quad \Gamma \Rightarrow \Delta, \psi}{\Gamma \Rightarrow \Delta, \varphi \wedge \psi}\ (\wedge R)$$

$$\frac{\varphi, \Gamma \Rightarrow \Delta \quad \psi, \Gamma \Rightarrow \Delta}{\varphi \vee \psi, \Gamma \Rightarrow \Delta}\ (\vee L) \qquad \frac{\Gamma \Rightarrow \Delta, \varphi}{\Gamma \Rightarrow \Delta, \varphi \vee \psi}\ (\vee R_1) \qquad \frac{\Gamma \Rightarrow \Delta, \psi}{\Gamma \Rightarrow \Delta, \varphi \vee \psi}\ (\vee R_2)$$

$$\frac{\Gamma \Rightarrow \Delta, \varphi \quad \psi, \Gamma \Rightarrow \Delta}{\varphi \rightarrow \psi, \Gamma \Rightarrow \Delta}\ (\rightarrow L) \qquad \frac{\varphi, \Gamma \Rightarrow \Delta, \psi}{\Gamma \Rightarrow \Delta, \varphi \rightarrow \psi}\ (\rightarrow R)$$

$$\frac{\Gamma \Rightarrow \Delta, \varphi}{\neg\varphi, \Gamma \Rightarrow \Delta}\ (\neg L) \qquad \frac{\varphi, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, \neg\varphi}\ (\neg R)$$

$$\frac{\varphi\left\{\frac{t}{x}\right\}, \Gamma \Rightarrow \Delta}{\forall x\varphi, \Gamma \Rightarrow \Delta}\ (\forall L) \qquad \frac{\Gamma \Rightarrow \Delta, \varphi\left\{\frac{y}{x}\right\}}{\Gamma \Rightarrow \Delta, \forall x\varphi}\ (\forall R)^*$$

$$\frac{\varphi\left\{\frac{y}{x}\right\}, \Gamma \Rightarrow \Delta}{\exists x\varphi, \Gamma \Rightarrow \Delta}\ (\exists L)^* \qquad \frac{\Gamma \Rightarrow \Delta, \varphi\left\{\frac{t}{x}\right\}}{\Gamma \Rightarrow \Delta, \exists x\varphi}\ (\exists R)$$

The System $\mathcal{LK}_=$ [Gentzen, '34]

$$\frac{\Gamma \Rightarrow \Delta}{\varphi, \Gamma \Rightarrow \Delta}\,(wkL) \qquad \frac{\Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, \varphi}\,(wkR)$$

$$\frac{\varphi, \varphi, \Gamma \Rightarrow \Delta}{\varphi, \Gamma \Rightarrow \Delta}\,(cntL) \qquad \frac{\Gamma \Rightarrow \Delta, \varphi, \varphi}{\Gamma \Rightarrow \Delta, \varphi}\,(cntR)$$

$$\frac{\Gamma \Rightarrow \Delta, \varphi \quad \varphi, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta}\,(cut) \qquad \frac{\Gamma \Rightarrow \Delta}{\Gamma\left\{\frac{\vec{s}}{\vec{x}}\right\} \Rightarrow \Delta\left\{\frac{\vec{s}}{\vec{x}}\right\}}\,(sub)$$

$$\frac{}{\varphi \Rightarrow \varphi}\,(id)$$

$$\frac{\Gamma \Rightarrow \Delta, s = t \quad \Gamma \Rightarrow \Delta, \varphi\left\{\frac{s}{x}\right\}}{\Gamma \Rightarrow \Delta, \varphi\left\{\frac{t}{x}\right\}}\,(eq) \qquad \frac{}{\Rightarrow t = t}\,(eq)$$

Finitary Proof System – RTC_G

Reflexivity

$$\Gamma \Rightarrow \Delta, (RTC_{x,y}\varphi)(s,s)$$

Step

$$\frac{\Gamma \Rightarrow \Delta, (RTC_{x,y}\varphi)(s,r) \quad \Gamma \Rightarrow \Delta, \varphi\left\{\frac{r}{x}, \frac{t}{y}\right\}}{\Gamma \Rightarrow \Delta, (TC_{x,y}\varphi)(s,t)}$$

Induction

$$\frac{\Gamma, \psi(x), \varphi(x,y) \Rightarrow \Delta, \psi\left\{\frac{y}{x}\right\}}{\Gamma, \psi\left\{\frac{s}{x}\right\}, (RTC_{x,y}\varphi)(s,t) \Rightarrow \Delta, \psi\left\{\frac{t}{x}\right\}}$$

provided $x \notin FV(\Gamma \cup \Delta)$ and $y \notin FV(\Gamma \cup \Delta \cup \{\psi\})$.

RTC_G 'Captures' TC-logic

$$\frac{\Gamma \Rightarrow \Delta, \varphi\left\{\frac{s}{x}, \frac{r}{y}\right\} \quad \Gamma \Rightarrow \Delta, (RTC_{x,y}\varphi)(r,t)}{\Gamma \Rightarrow \Delta, (RTC_{x,y}\varphi)(s,t)}$$

$$\frac{\Gamma \Rightarrow \Delta, (RTC_{x,y}\varphi)(s,t)}{\Gamma \Rightarrow \Delta, (RTC_{y,x}\varphi)(t,s)}$$

$$\frac{\Gamma, \varphi \Rightarrow \Delta, \psi}{\Gamma, (RTC_{x,y}\varphi)(s,t) \Rightarrow \Delta, (RTC_{x,y}\psi)(s,t)}$$

$$\frac{\Gamma \Rightarrow \Delta, (RTC_{x,y}\varphi)(s,t)}{\Gamma \Rightarrow \Delta, \left(RTC_{u,v}\varphi\left\{\frac{u}{x}, \frac{v}{y}\right\}\right)(s,t)}$$

$$\frac{(RTC_{x,y}\varphi)(s,t), \Gamma \Rightarrow \Delta}{(RTC_{u,v}(RTC_{x,y}\varphi)(u,v))(s,t), \Gamma \Rightarrow \Delta}$$

$$\frac{\varphi\left\{\frac{s}{x}\right\}, \Gamma \Rightarrow \Delta}{(RTC_{x,y}\varphi)(s,t), \Gamma \Rightarrow s = t, \Delta}$$

$$\frac{\Gamma \Rightarrow \Delta, (RTC_{x,y}\varphi)(s,t)}{\Gamma \Rightarrow \Delta, s = t, \exists z\left((RTC_{x,y}\varphi)(s,z) \wedge \varphi\left\{\frac{z}{x}, \frac{t}{y}\right\}\right)}$$

Arithmetics in RTC_G

TC for Arithmetics

RTC_G+A is obtained from RTC_G by the addition of the standard axioms for successor and addition, and the axiom characterizing the natural numbers in TC-logic.

Arithmetics in RTC_G

TC for Arithmetics

RTC_G+A is obtained from RTC_G by the addition of the standard axioms for successor and addition, and the axiom characterizing the natural numbers in TC-logic.

Theorem

RTC_G+A is equivalent to the sequent calculi of *PA*, i.e. there is a provability preserving translation algorithm between them.

Arithmetics in RTC_G

TC for Arithmetics

RTC_G+A is obtained from RTC_G by the addition of the standard axioms for successor and addition, and the axiom characterizing the natural numbers in TC-logic.

Theorem

RTC_G+A is equivalent to the sequent calculi of *PA*, i.e. there is a provability preserving translation algorithm between them.

Corollary

The ordinal number of the RTC_G+A is ε_0.

Henkin Semantics

A σ-**Henkin structure** is a triple $M = \langle D, I, D' \rangle$ (frame), s.t.:

1. $\langle D, I \rangle$ is a FO structure for σ
2. $D' \subseteq P(D)$ is closed under parametric definability.

Henkin Semantics

A σ-**Henkin structure** is a triple $M = \langle D, I, D' \rangle$ (frame), s.t.:

1. $\langle D, I \rangle$ is a FO structure for σ
2. $D' \subseteq P(D)$ is closed under parametric definability.

$M, v \models (RTC_{x,y}\varphi)(s, t)$ provided for every $A \in D'$, if $v(s) \in A$ and $\forall a, b \in D: (a \in A \wedge M, v[x := a, y := b] \models \varphi) \rightarrow b \in A$, then $v(t) \in A$.

Henkin Semantics

A σ-**Henkin structure** is a triple $M = \langle D, I, D' \rangle$ (frame), s.t.:

1. $\langle D, I \rangle$ is a FO structure for σ
2. $D' \subseteq P(D)$ is closed under parametric definability.

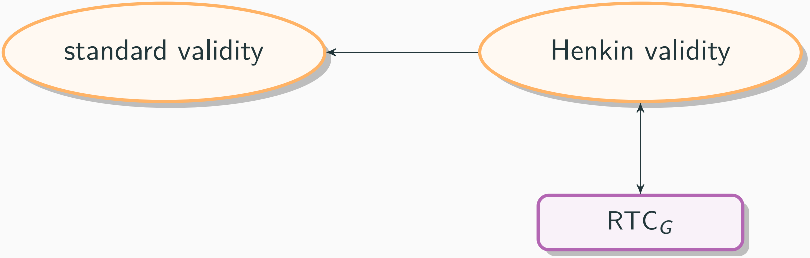
$M, v \models (RTC_{x,y}\varphi)(s,t)$ provided for every $A \in D'$, if $v(s) \in A$ and $\forall a, b \in D : (a \in A \wedge M, v[x := a, y := b] \models \varphi) \rightarrow b \in A$, then $v(t) \in A$.

Completeness Theorem

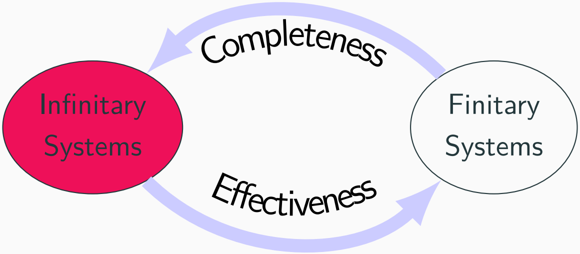
$T \vdash_{\mathrm{RTC}_G} \varphi \iff T \models_H \varphi$.

So Far

So Far



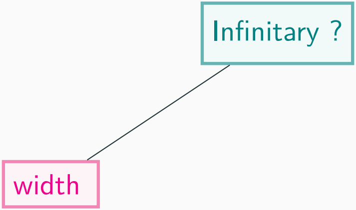
Proof Theory



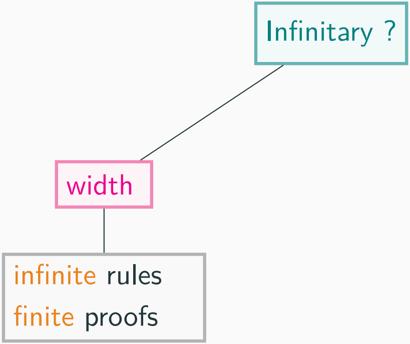
Infinitary Systems

Infinitary ?

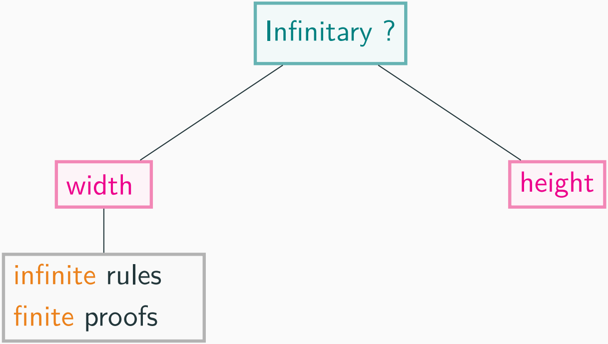
Infinitary Systems



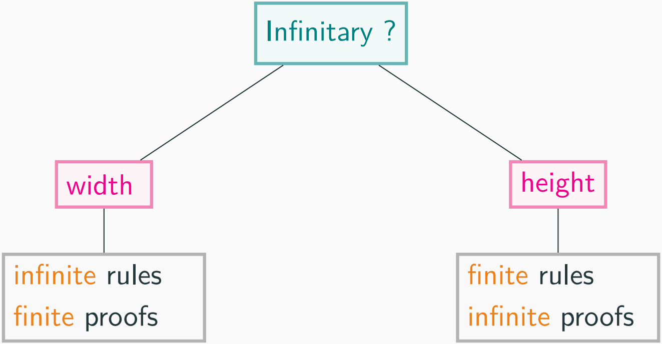
Infinitary Systems



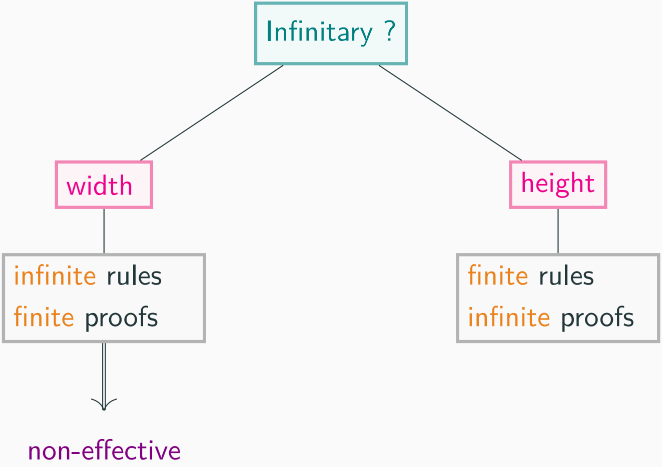
Infinitary Systems



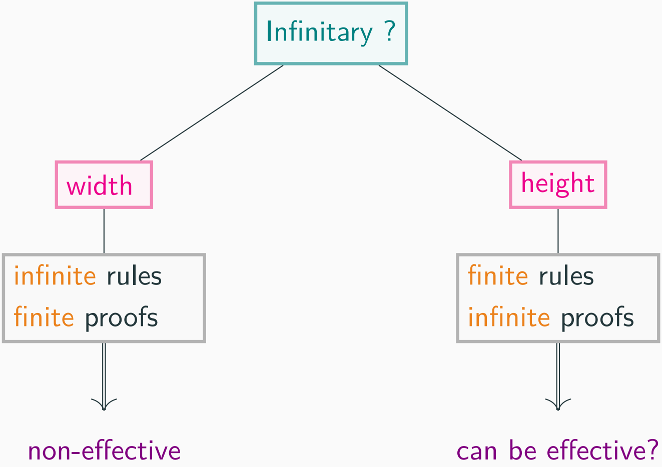
Infinitary Systems



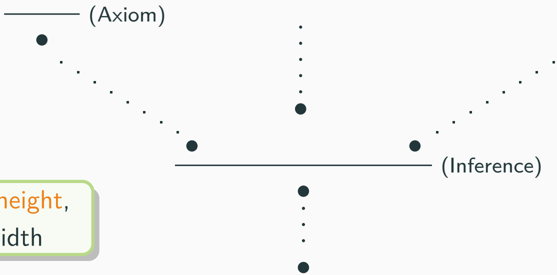
Infinitary Systems



Infinitary Systems

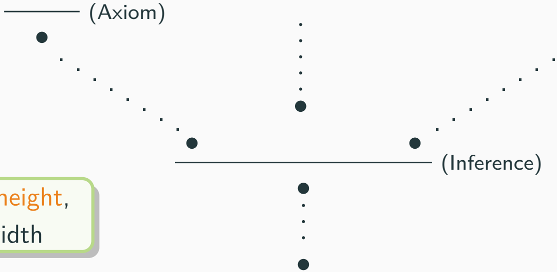


Infinite Descent-Style Proof System



Infinite height, not width

Infinite Descent-Style Proof System



Infinite height, not width

- Proofs can be infinite, non-well-founded trees, provided that every infinite path admits some infinite descent.
- The descent is witnessed by tracing terms/formulas corresponding to elements of a well-founded set.
- This global trace condition is decidable using Büchi automata.
- Systems of implicit induction.

Infinitary Proof System – RTC_G^ω

Reflexivity

$$\Gamma \Rightarrow \Delta, (RTC_{x,y}\varphi)\,(s, s)$$

Step

$$\frac{\Gamma \Rightarrow \Delta, (RTC_{x,y}\varphi)\,(s, r) \quad \Gamma \Rightarrow \Delta, \varphi\left\{\frac{r}{x}, \frac{t}{y}\right\}}{\Gamma \Rightarrow \Delta, (TC_{x,y}\varphi)\,(s, t)}$$

Case-split

$$\frac{\Gamma, s = t \Rightarrow \Delta \qquad \Gamma, (RTC_{x,y}\varphi)(s, z), \varphi\left\{\frac{z}{x}, \frac{t}{y}\right\} \Rightarrow \Delta}{\Gamma, (RTC_{x,y}\varphi)(s, t) \Rightarrow \Delta}$$

provided z is fresh.

Infinitary Proof System – RTC_G^ω

Reflexivity

$$\Gamma \Rightarrow \Delta, (RTC_{x,y}\varphi)\,(s, s)$$

Step

$$\frac{\Gamma \Rightarrow \Delta, (RTC_{x,y}\varphi)\,(s, r) \quad \Gamma \Rightarrow \Delta, \varphi\left\{\frac{r}{x}, \frac{t}{y}\right\}}{\Gamma \Rightarrow \Delta, (TC_{x,y}\varphi)\,(s, t)}$$

Case-split

$$\frac{\Gamma, s = t \Rightarrow \Delta \qquad \Gamma, (RTC_{x,y}\varphi)(s, z), \varphi\left\{\frac{z}{x}, \frac{t}{y}\right\} \Rightarrow \Delta}{\Gamma, (RTC_{x,y}\varphi)(s, t) \Rightarrow \Delta}$$

provided z is fresh.

Soundness and Completeness

Completeness Theorem

$T \vdash^{cf}_{\mathsf{RTC}^{\omega}_{G}} \varphi \iff T \models \varphi.$

Soundness and Completeness

Completeness Theorem

$T \vdash^{cf}_{\mathsf{RTC}^{\omega}_{G}} \varphi \Longleftrightarrow T \models \varphi.$

Global soundness via an infinite descent proof-by-contradiction:

Soundness and Completeness

Completeness Theorem

$T \vdash^{cf}_{\mathsf{RTC}^{\omega}_{G}} \varphi \iff T \models \varphi.$

Global soundness via an infinite descent proof-by-contradiction:

- Assume the conclusion of the proof is invalid

Soundness and Completeness

Completeness Theorem

$T \vdash^{cf}_{\mathrm{RTC}^{\omega}_{G}} \varphi \Longleftrightarrow T \models \varphi.$

Global soundness via an infinite descent proof-by-contradiction:

- Assume the conclusion of the proof is invalid
- Local soundness entails an infinite sequence of counter models

Soundness and Completeness

Completeness Theorem

$T \vdash^{cf}_{\mathsf{RTC}^{\omega}_{G}} \varphi \Longleftrightarrow T \models \varphi.$

Global soundness via an infinite descent proof-by-contradiction:

- Assume the conclusion of the proof is invalid
- Local soundness entails an infinite sequence of counter models
    - Mapped to the minimal length for witnessing the transitive closure trace.

Soundness and Completeness

Completeness Theorem

$T \vdash^{cf}_{\mathrm{RTC}^{\omega}_{G}} \varphi \iff T \models \varphi.$

Global soundness via an infinite descent proof-by-contradiction:

- Assume the conclusion of the proof is invalid
- Local soundness entails an infinite sequence of counter models
    - Mapped to the minimal length for witnessing the transitive closure trace.
- Global trace condition entails the chain is infinitely descending

Soundness and Completeness

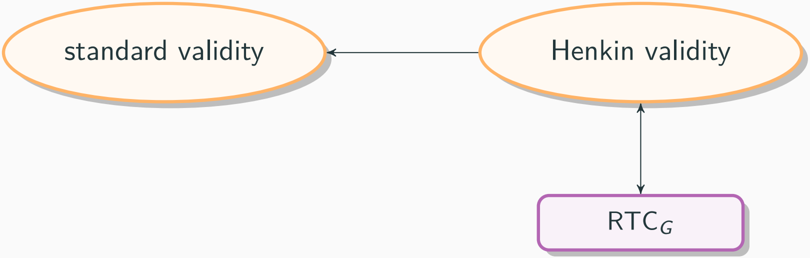
Completeness Theorem

$T \vdash^{cf}_{\mathsf{RTC}^{\omega}_{G}} \varphi \Longleftrightarrow T \models \varphi.$

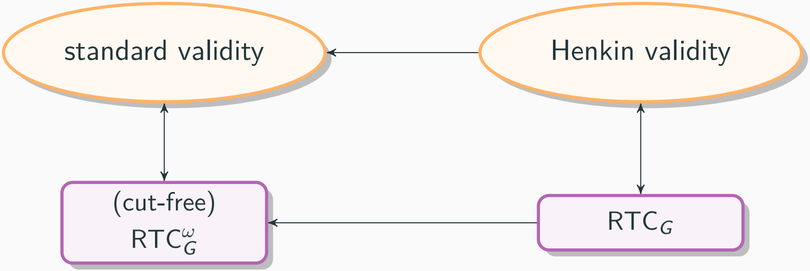
Global soundness via an infinite descent proof-by-contradiction:

- Assume the conclusion of the proof is invalid
- Local soundness entails an infinite sequence of counter models
  - Mapped to the minimal length for witnessing the transitive closure trace.
- Global trace condition entails the chain is infinitely descending
  - But the numbers are well-founded ... contradiction!

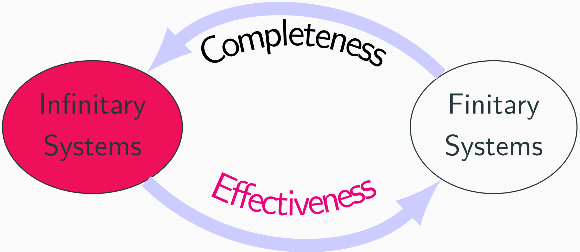
So Far



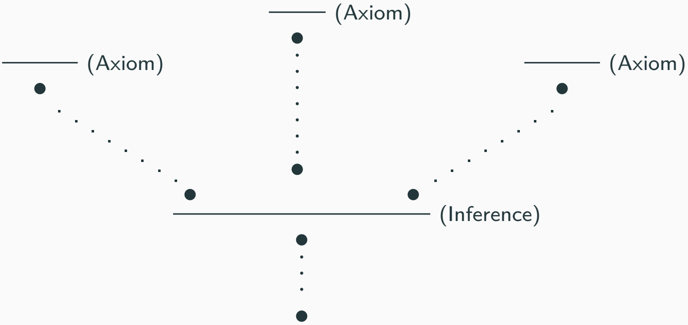
So Far


standard validity
Henkin validity
(cut-free)
RTC^{ω}_{G}
RTC_G

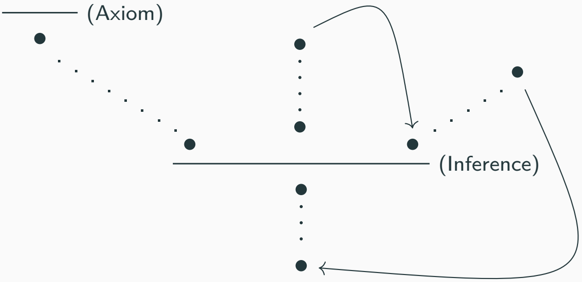
Proof Theory



The Cyclic Subsystem – CRTC_G^ω



The Cyclic Subsystem – CRTC_G^ω



- An effective subsystem can be obtained by considering only the regular infinite proofs.
- Regular proofs = represented as finite, possibly cyclic, graphs.

Implicit Induction Subsumes Explicit Induction

$$\frac{\Gamma, \psi\left\{\frac{v}{x}\right\}, (RTC_{x,y}\,\varphi)(v,w) \Rightarrow \Delta, \psi\left\{\frac{w}{x}\right\}}{\Gamma, \psi\left\{\frac{v}{x}\right\}, (RTC_{x,y}\,\varphi)(v,z) \Rightarrow \Delta, \psi\left\{\frac{z}{x}\right\}}\ \text{(Subst)}$$

$$\frac{\Gamma, \psi(x), \varphi(x,y) \Rightarrow \Delta, \psi\left\{\frac{y}{x}\right\}}{\Gamma, \psi\left\{\frac{z}{x}\right\}, \varphi\left\{\frac{z}{x}, \frac{w}{y}\right\} \Rightarrow \Delta, \psi\left\{\frac{w}{x}\right\}}\ \text{(Subst)}$$

(Cut)

$$\Gamma, \psi\left\{\frac{v}{x}\right\}, (RTC_{x,y}\,\varphi)(v,z), \varphi\left\{\frac{z}{x}, \frac{w}{y}\right\} \Rightarrow \Delta, \psi\left\{\frac{w}{x}\right\}$$

$$\vdots$$

$$\frac{}{\psi\left\{\frac{v}{x}\right\}, v = w \Rightarrow \psi\left\{\frac{w}{x}\right\}}\ \text{(Eq)}$$

(Case-split)

$$\frac{\Gamma, \psi\left\{\frac{v}{x}\right\}, (RTC_{x,y}\,\varphi)(v,w) \Rightarrow \Delta, \psi\left\{\frac{w}{x}\right\}}{\Gamma, \psi\left\{\frac{s}{x}\right\}, (RTC_{x,y}\,\varphi)(s,t) \Rightarrow \Delta, \psi\left\{\frac{t}{x}\right\}}\ \text{(Subst)}$$

Implicit Induction Subsumes Explicit Induction

$$
\dfrac{\Gamma, \psi\left\{\frac{v}{x}\right\}, (RTC_{x,y}\,\varphi)(v, w) \Rightarrow \Delta, \psi\left\{\frac{w}{x}\right\}}{\Gamma, \psi\left\{\frac{v}{x}\right\}, (RTC_{x,y}\,\varphi)(v, z) \Rightarrow \Delta, \psi\left\{\frac{z}{x}\right\}}\ \text{(Subst)}
\qquad
\dfrac{\Gamma, \psi(x), \varphi(x, y) \Rightarrow \Delta, \psi\left\{\frac{y}{x}\right\}}{\Gamma, \psi\left\{\frac{z}{x}\right\}, \varphi\left\{\frac{z}{x}, \frac{w}{y}\right\} \Rightarrow \Delta, \psi\left\{\frac{w}{x}\right\}}\ \text{(Subst)}
$$

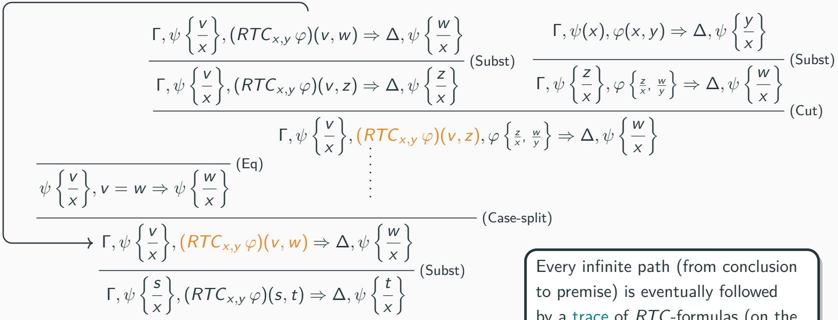
$$
\dfrac{\qquad\qquad}{\Gamma, \psi\left\{\frac{v}{x}\right\}, (RTC_{x,y}\,\varphi)(v, z), \varphi\left\{\frac{z}{x}, \frac{w}{y}\right\} \Rightarrow \Delta, \psi\left\{\frac{w}{x}\right\}}\ \text{(Cut)}
$$

$$
\dfrac{}{\psi\left\{\frac{v}{x}\right\}, v = w \Rightarrow \psi\left\{\frac{w}{x}\right\}}\ \text{(Eq)}
\qquad \vdots
$$

$$
\dfrac{\dfrac{\qquad\qquad}{\Gamma, \psi\left\{\frac{v}{x}\right\}, (RTC_{x,y}\,\varphi)(v, w) \Rightarrow \Delta, \psi\left\{\frac{w}{x}\right\}}\ \text{(Case-split)}}{\Gamma, \psi\left\{\frac{s}{x}\right\}, (RTC_{x,y}\,\varphi)(s, t) \Rightarrow \Delta, \psi\left\{\frac{t}{x}\right\}}\ \text{(Subst)}
$$

Every infinite path (from conclusion to premise) is eventually followed by a trace of *RTC*-formulas (on the left-hand side) which progresses (via case-split) infinitely often.

Implicit Induction Subsumes Explicit Induction



Every infinite path (from conclusion to premise) is eventually followed by a trace of *RTC*-formulas (on the left-hand side) which progresses (via case-split) infinitely often.

- Normal Cyclic Proofs = non-overlapping cyclic proofs.

Cyclic Proof vs. Explicit Induction

Induction invariant

Cyclic Proof vs. Explicit Induction

Induction invariant

Explicit induction requires it *a priori*

Major challenge for automatic proof search

Cyclic Proof vs. Explicit Induction

Induction invariant

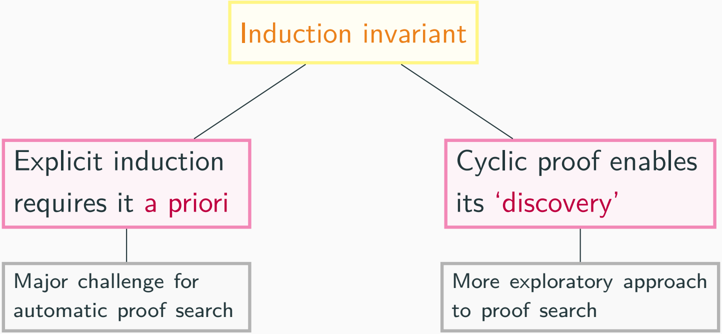
Explicit induction requires it *a priori*

Cyclic proof enables its 'discovery'

Major challenge for automatic proof search

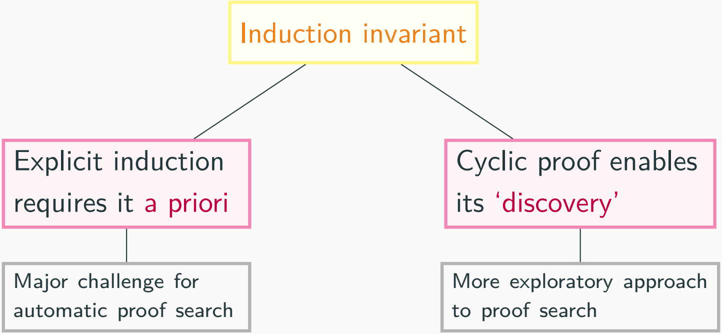
More exploratory approach to proof search

Cyclic Proof vs. Explicit Induction



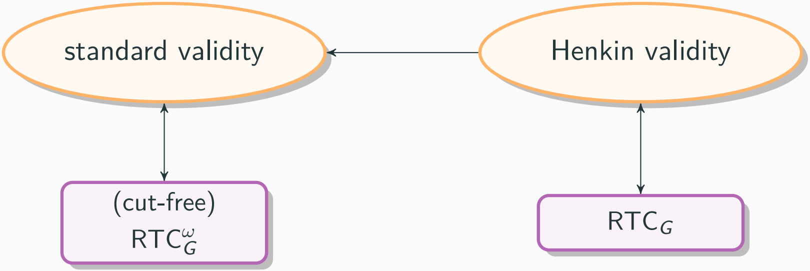
- Complex induction schemes naturally represented by nested and overlapping cycles.

Cyclic Proof vs. Explicit Induction



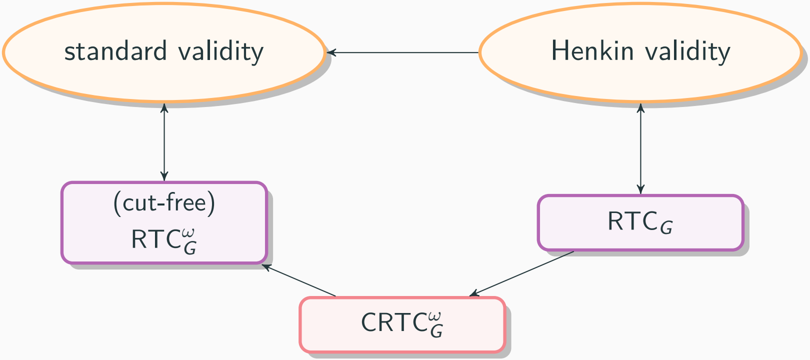
- Complex induction schemes naturally represented by nested and overlapping cycles.
- Every sequent provable using the explicit induction rule is also derivable using cyclic proof.

So Far

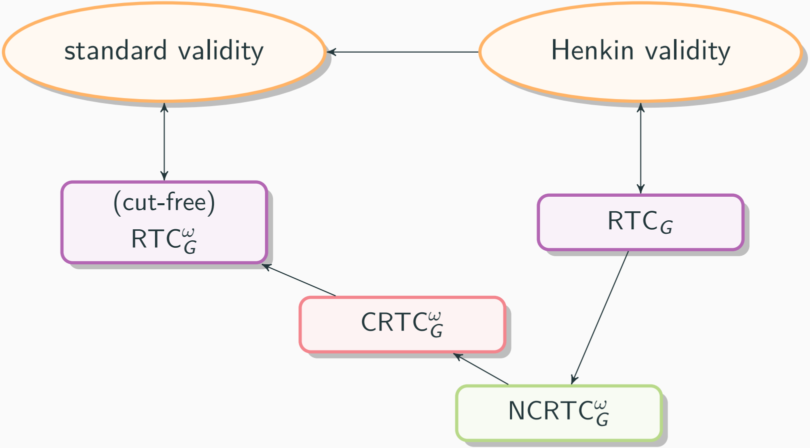


standard validity
Henkin validity
(cut-free)
RTC_G^ω
RTC_G

So Far


standard validity
Henkin validity
(cut-free)
RTC^{ω}_{G}
RTC_G
$CRTC^{\omega}_{G}$

So Far


standard validity
Henkin validity
(cut-free)
RTC_G^ω
RTC_G
$CRTC_G^\omega$
$NCRTC_G^\omega$

Is the Cyclic System Stronger?

- For arithmetics, the explicit and cyclic systems are equivalent.

Is the Cyclic System Stronger?

- For arithmetics, the explicit and cyclic systems are equivalent.
- In general, the question of the (in)equivalence between the systems remains open.

Is the Cyclic System Stronger?

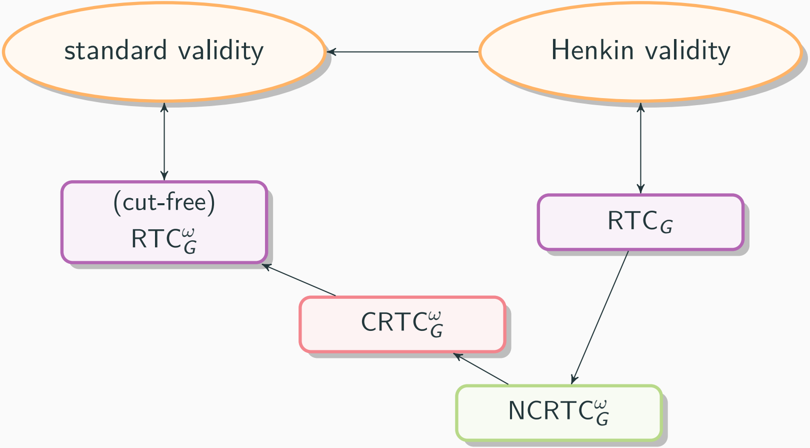
- For arithmetics, the explicit and cyclic systems are equivalent.
- In general, the question of the (in)equivalence between the systems remains open.
- In systems for FOL with inductive definition, the equivalence was refuted when both systems have the same set of inductive definitions. [Berardi, Tatsuta, 2017]

Is the Cyclic System Stronger?

- For arithmetics, the explicit and cyclic systems are equivalent.
- In general, the question of the (in)equivalence between the systems remains open.
- In systems for FOL with inductive definition, the equivalence was refuted when both systems have the same set of inductive definitions. [Berardi, Tatsuta, 2017]

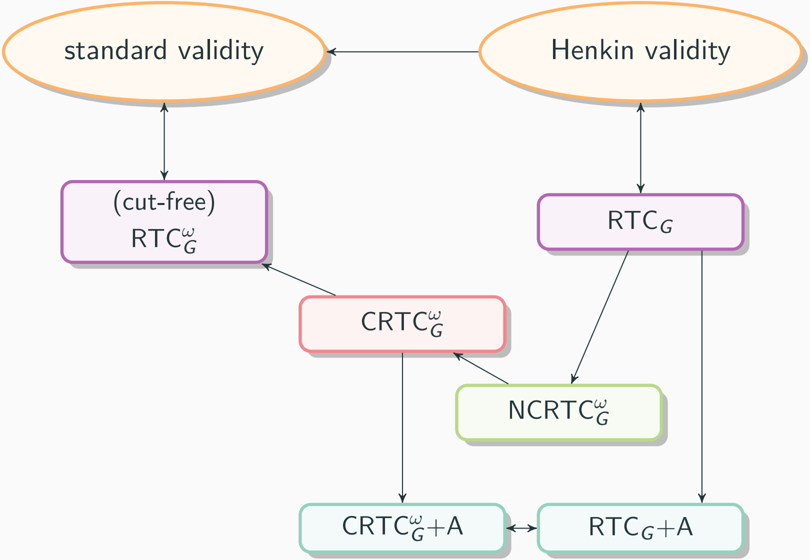
- In the TC framework all inductive definitions at once.

So Far



standard validity
Henkin validity
(cut-free)
RTC_G^ω
RTC_G
$CRTC_G^\omega$
$NCRTC_G^\omega$

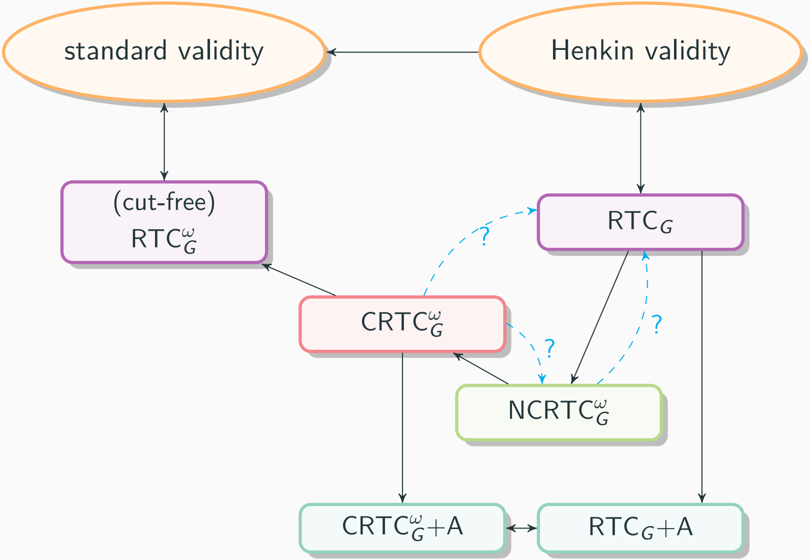
So Far


standard validity
Henkin validity
(cut-free)
RTC^{ω}_{G}
RTC_{G}
$CRTC^{\omega}_{G}$
$NCRTC^{\omega}_{G}$
$CRTC^{\omega}_{G}$+A
RTC_{G}+A

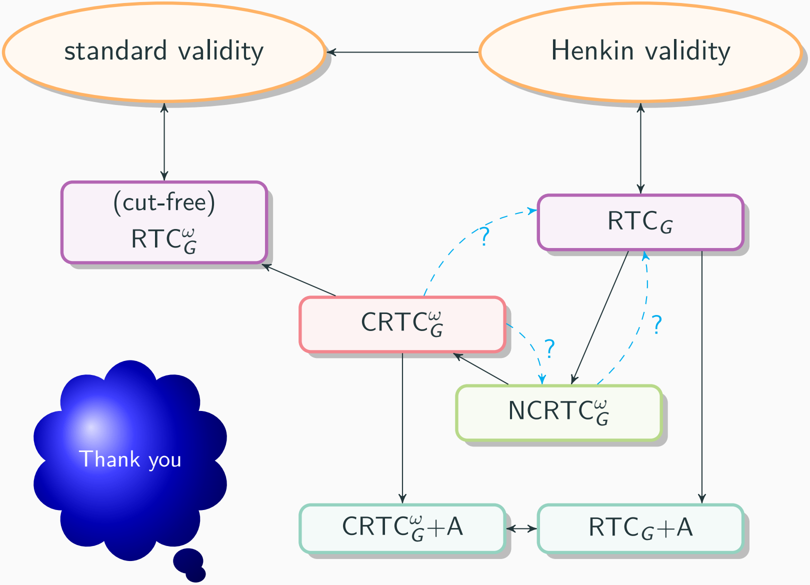
Future (and Current) Work

- Resolving the open question of the (in)equivalence of RTC_G and CRTC_G^ω.
- Implementing CRTC_G^ω and investigating the practicalities of TC-logic to support automated inductive reasoning.
- Using the uniformity of TC-logic to better study the relationship between implicit and explicit induction.
    - Cuts required in each system
    - Relative complexity of proofs
- Incorporating coinductive reasoning into the formal system.

Summary


standard validity
Henkin validity
(cut-free)
RTC^{ω}_{G}
RTC_{G}
?
$CRTC^{\omega}_{G}$
?
?
$NCRTC^{\omega}_{G}$
$CRTC^{\omega}_{G}$+A
RTC_{G}+A

Summary


standard validity
Henkin validity
(cut-free)
RTC_G^ω
RTC_G
$CRTC_G^\omega$
?
?
?
$NCRTC_G^\omega$
$CRTC_G^\omega$+A
RTC_G+A
Thank you