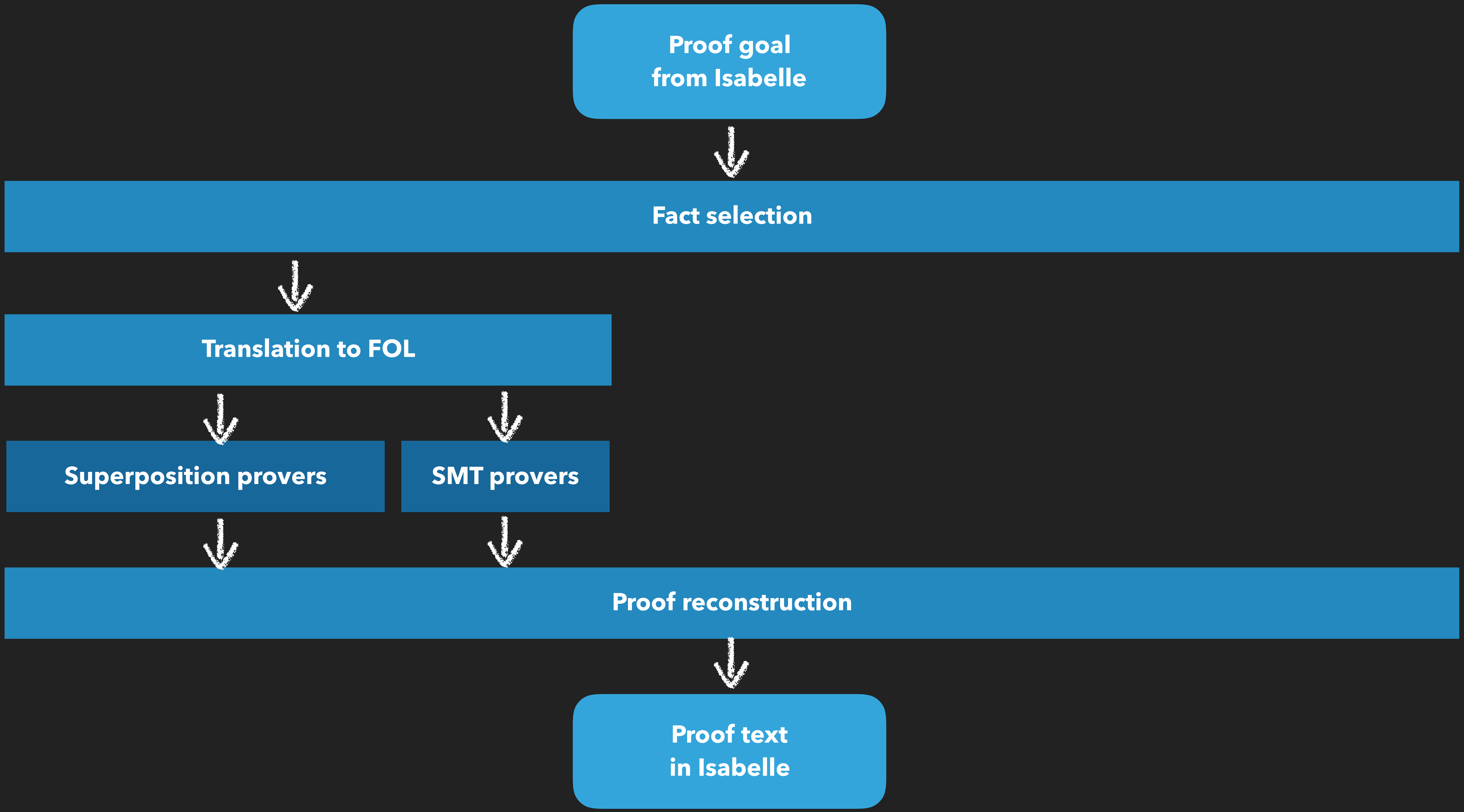
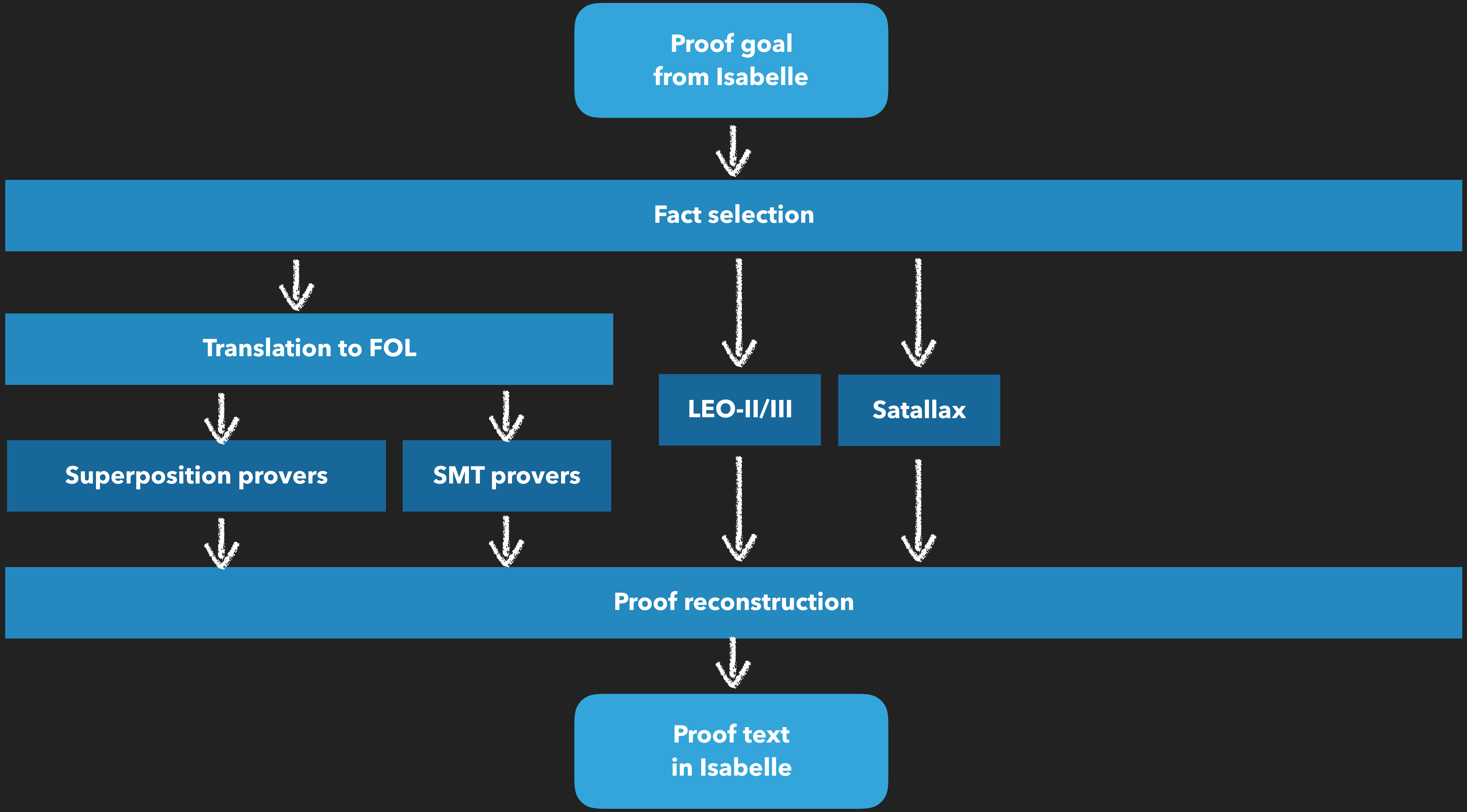
ALEXANDER BENTKAMP
JASMIN BLANCHETTE
SIMON CRUANES
UWE WALDMANN

SUPERPOSITION FOR LAMBDA-FREE HIGHER-ORDER LOGIC

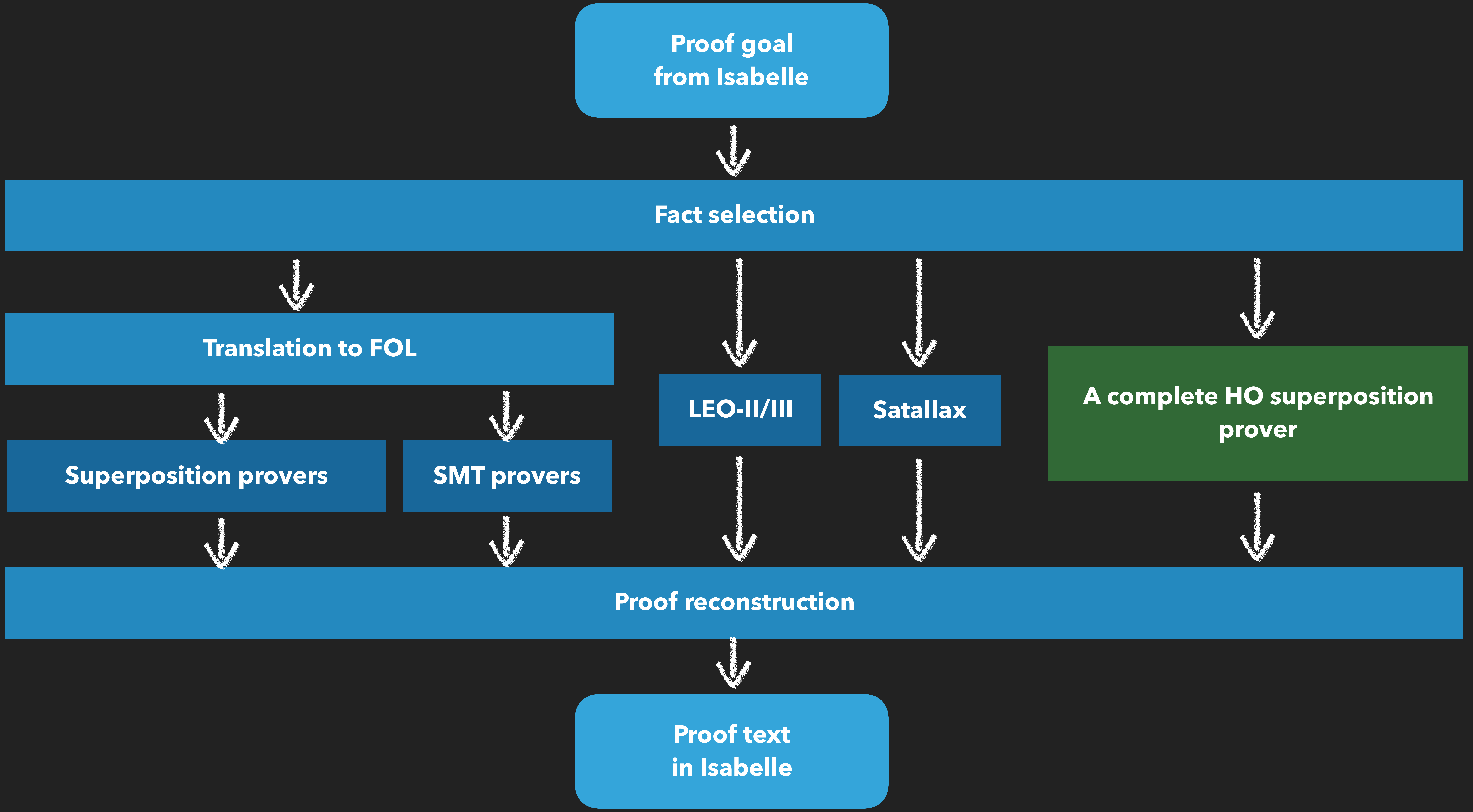
Motivation: Sledgehammer



Motivation: Sledgehammer



Motivation: Sledgehammer



DESIGN PRINCIPLE: BE GRACEFUL

HO superposition on first-order problems should coincide with FO superposition

Our way to higher-order superposition

Translation to FOL: applicative encoding

$f\ (H\ f)$ is translated to $app(f, app(H, f))$

λ-free HOL — FOL

Translation to FOL: applicative encoding

f $(H$ f$)$ is translated to app(f, app$(H$, f$)$)

λ-free HOL — FOL

NOT GRACEFUL!

Compatibility with arguments?

$t > s \Rightarrow t\, u > s\, u$

Term orders for λ-free HOL

Compatibility with arguments?

$t > s \Rightarrow t\ u > s\ u$

KBO without argument coefficients

Yes:
Completeness proof works as in FOL

Term orders for λ-free HOL

Compatibility with arguments?

$t > s \Rightarrow t\ u > s\ u$

KBO without argument coefficients

LPO

KBO with argument coefficients

Yes:
Completeness proof works as in FOL

No:
This is the topic of my talk

The superposition rule

$$\frac{D \vee t = t' \qquad\qquad C \vee (\neg)\, s[u] = s'}{(D \vee C \vee (\neg)\, s[t'] = s')\sigma}\ \sigma = \mathrm{mgu}(t,u)$$

+ order conditions

Superposition only into argument subterms

Argument subterms: f a (h b c)

Prefix subterms: f a (h b c)

Superposition only into argument subterms

Argument subterms: $\underline{\mathrm{f}\ \underline{\mathrm{a}}\ (\underline{\underline{\mathrm{h}\ \underline{\mathrm{b}}\ \underline{\mathrm{c}}}})}$

Prefix subterms: $\underline{\underline{\mathrm{f}}\ \mathrm{a}}\ (\underline{\underline{\mathrm{h}}\ \mathrm{b}}\ \mathrm{c})$

$$\frac{\mathrm{g} = \mathrm{f} \qquad\qquad \mathrm{g\ a} \neq \mathrm{b}}{\mathrm{f\ a} \neq \mathrm{b}}\ \text{~~Sup~~}$$

Argument congruence rule

$$\frac{C \vee t = s}{C \vee t\,X = s\,X}\ \textsc{ArgCong}$$

Argument congruence rule

$$\frac{C \vee t = s}{C \vee t\,X = s\,X}\ \textsc{ArgCong}$$

Example:

$$\textsc{Sup}\ \frac{\textsc{ArgCong}\ \dfrac{g = f}{g\,X = f\,X} \qquad g\,a \neq b}{f\,a \neq b}$$

Argument congruence rule

$$\frac{C \vee t = s}{C \vee t\,X = s\,X}\ \textsc{ArgCong}$$

BUT ISN'T THIS RULE ALWAYS REDUNDANT?

Floor encoding

Encode **ground λ-free HOL** terms into **FOL:**

$$\lfloor f \rfloor = f_0$$
$$\lfloor f\ a \rfloor = f_1(a_0)$$

Redundancy is defined with respect to this encoding.

Example:

$$\textsc{ArgCong}\ \frac{g = f}{g\ X = f\ X}$$

$$\frac{g_0 = f_0}{g_1\ a_0 = f_1\ a_0}$$

Not redundant!

What changes in the proof?

Refutational completeness:

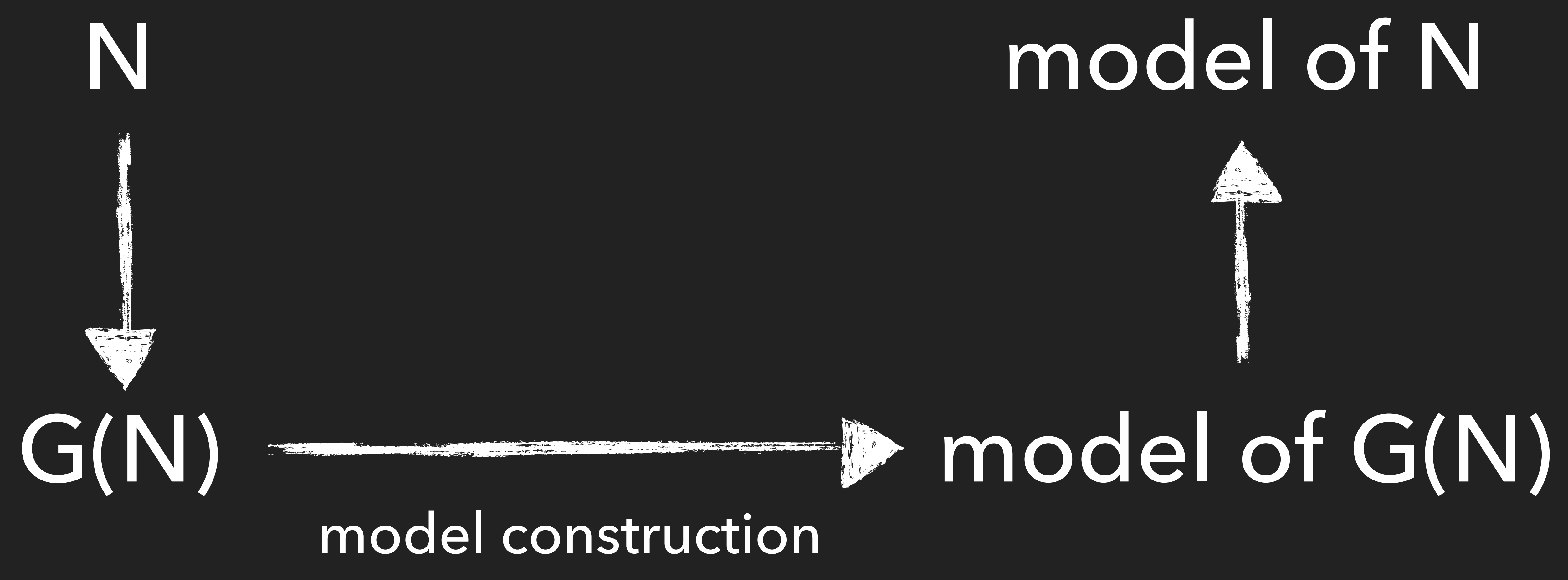
Let N be saturated up to redundancy, $\bot \notin N$.

Then N has a model.

What changes in the proof?

Refutational completeness:
Let N be saturated up to redundancy, $\bot \notin N$.
Then N has a model.

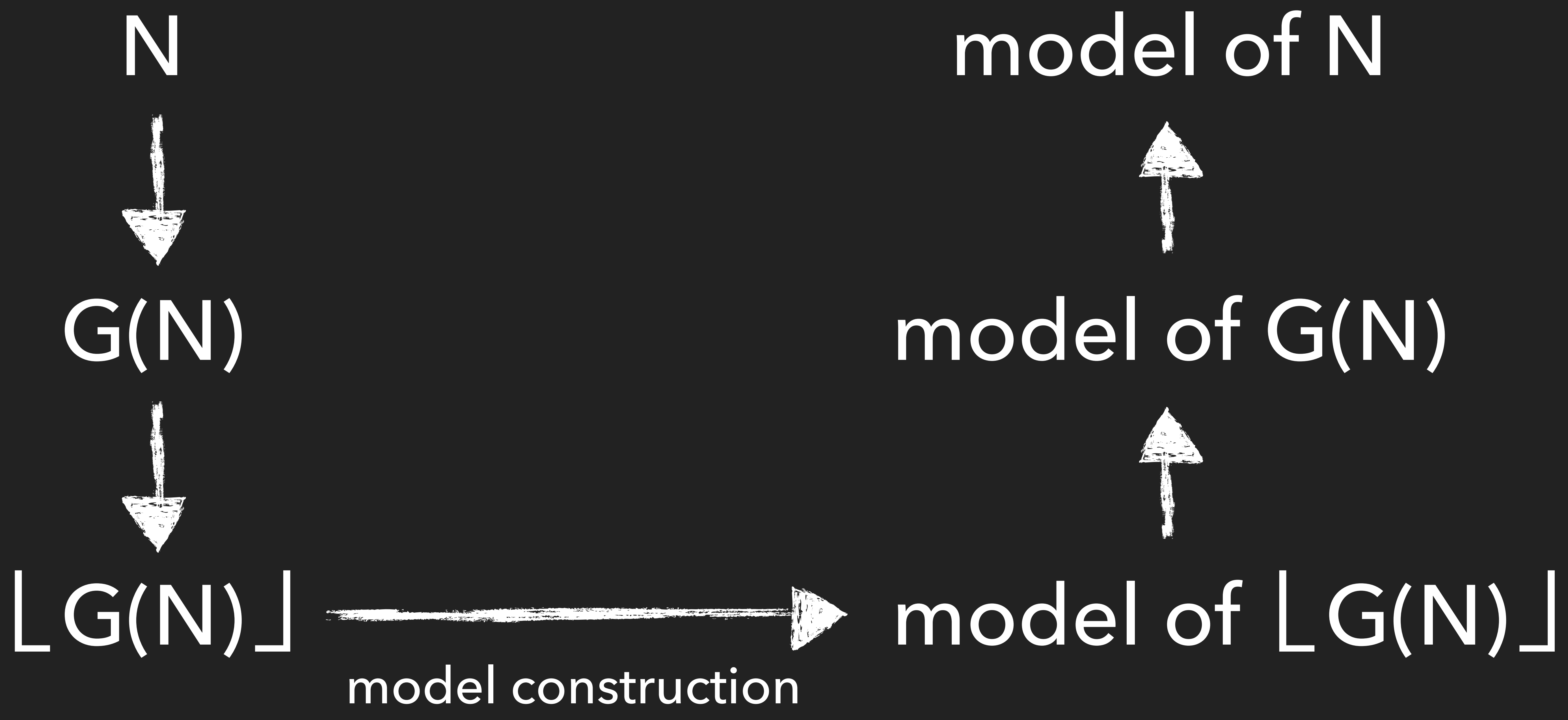
Proof sketch for FOL:



What changes in the proof?

Refutational completeness:
Let N be saturated up to redundancy, $\bot \notin N$.
Then N has a model.

Proof sketch for λ-free HOL:



Issue: superposition into variables

Example:

$$C = \ldots X \ldots X\ a \ldots$$

Given $g > f$, it is unclear whether $X := g$ or $X := f$ will yield the smaller clause

Issue: superposition into variables

Example:

$C = \ldots X \ldots X\, a \ldots$

Given $g > f$, it is unclear whether $X := g$ or $X := f$ will yield the smaller clause

Solution #1:
purifying calculus

$\ldots X\, \bar{u} \ldots X\, \bar{v} \ldots$

is purified to

$\ldots X\, \bar{u} \ldots Y\, \bar{v} \ldots \lor X \neq Y$

if $\bar{u} \neq \bar{v}$

Example:

$$C = \ldots X \ldots X\, a \ldots$$

Given $g > f$, it is unclear whether $X := g$ or $X := f$ will yield the smaller clause

Solution #1:
purifying calculus

$$\ldots X\, \bar{u} \ldots X\, \bar{v} \ldots$$

is purified to

$$\ldots X\, \bar{u} \ldots Y\, \bar{v} \ldots \vee X \neq Y$$

if $\bar{u} \neq \bar{v}$

Solution #2:
nonpurifying calculus

Perform superpositions at variables if the order situation is unclear

Evaluation of our prototype

using the Zipperposition theorem prover

	TPTP benchmarks		Judgment Day λ-free HOL benchmarks	
# unsat	FO	HO	32 facts	512 facts
first-order mode	**181**	-	-	-
applicative encoding	151	**677**	**873**	843
purifying calculus	180	647	851	**908**
nonpurifying calculus	179	669	866	889

Evaluation of our prototype

using the Zipperposition theorem prover

	TPTP benchmarks		Judgment Day λ-free HOL benchmarks	
# unsat	FO	HO	32 facts	512 facts
first-order mode	**181**	-	-	-
applicative encoding	151	**677**	**873**	843
purifying calculus	180	647	851	**908**
nonpurifying calculus	179	669	866	889

Evaluation of our prototype

using the Zipperposition theorem prover

# unsat	TPTP benchmarks		Judgment Day λ-free HOL benchmarks	
	FO	HO	32 facts	512 facts
first-order mode	**181**	-	-	-
applicative encoding	151	**677**	**873**	843
purifying calculus	180	647	851	**908**
nonpurifying calculus	179	669	866	889

Evaluation of our prototype

using the Zipperposition theorem prover

# unsat	TPTP benchmarks FO	TPTP benchmarks HO	Judgment Day λ-free HOL benchmarks 32 facts	Judgment Day λ-free HOL benchmarks 512 facts
first-order mode	**181**	-	-	-
applicative encoding	151	**677**	**873**	843
purifying calculus	180	647	851	**908**
nonpurifying calculus	179	669	866	889

Evaluation of our prototype

using the Zipperposition theorem prover

	TPTP benchmarks		Judgment Day λ-free HOL benchmarks	
# unsat	FO	HO	32 facts	512 facts
first-order mode	**181**	-	-	-
applicative encoding	151	**677**	**873**	843
purifying calculus	180	647	851	**908**
nonpurifying calculus	179	669	866	889

In summary

- We developed refutationally complete calculi for λ-free HOL
- They reduce the gap between HO proof assistants and superposition provers
- They are promising as a stepping stone towards a HO superposition calculus