Justifying the HOL-Nuprl Connection in the Categorical Framework of General Logics

José Meseguer   Mark-Oliver Stehr

SRI International

Acknowledgements

We thank:

- Robert Constable (Cornell) and Douglas Howe (Bell Labs) for sharing with us their ideas and implementation details about the HOL-Nurpl translation, and
- Carolyn Talcott (Stanford) and Francisco Durán, Steven Eker, and all our colleagues in the Maude Group at SRI for many useful conversations on the use of Maude and formal interoperability.

Motivation

Formal developments often require several man-months of effort and it is highly desirable that their results can be shared between different systems.

$\Rightarrow$ Here formal interoperability provides a *solution*.

The technology is evolving and there will never be a single tool that covers all the needs of formal verification.

$\Rightarrow$ Here formal interoperability is a *requirement*.

Both arguments apply to the connection between HOL and Nuprl.

HOL/Nuprl Overview

HOL

- based on *classical higher-oder logic*
- one of the most popular theorem provers
- a huge amount of mathematics has been developed

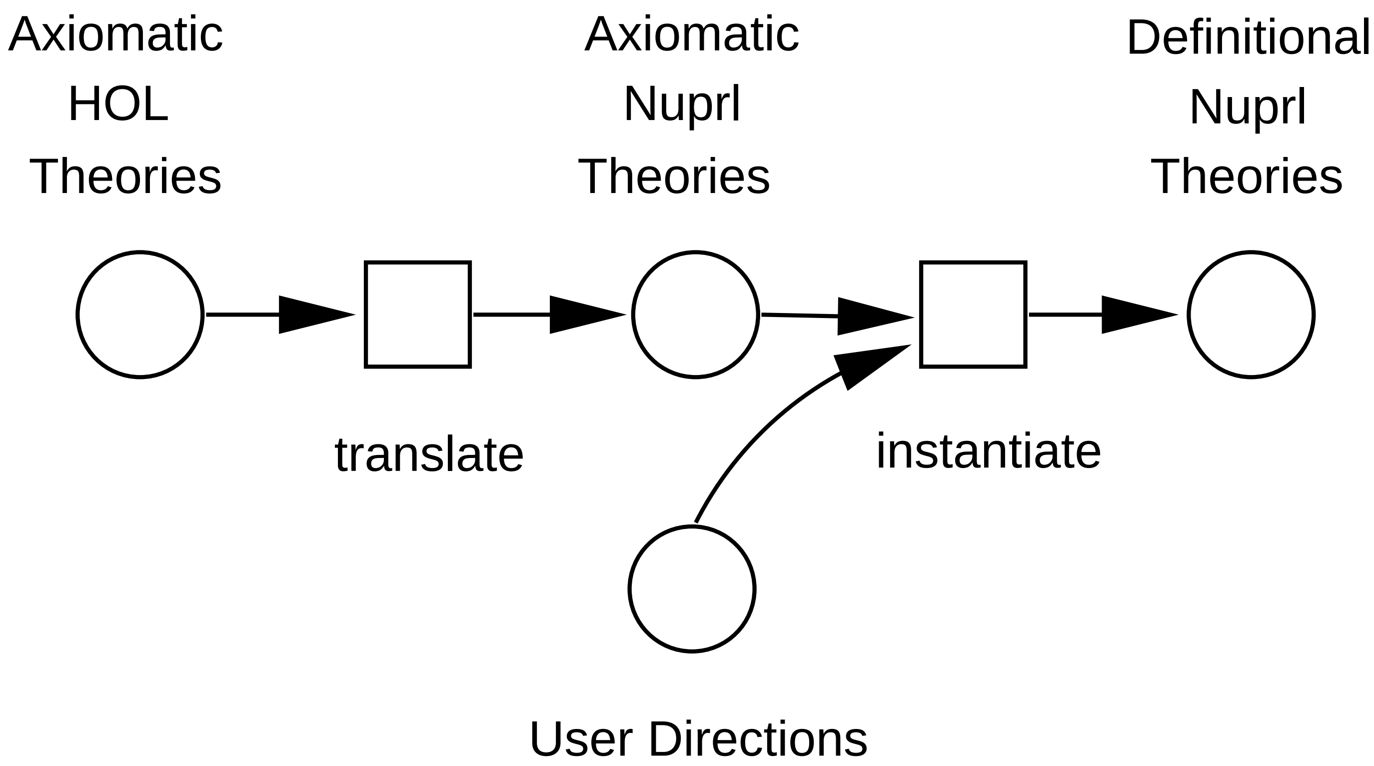
Nuprl

- descendent of Martin Loef's polymorphic type theory
- much more expressive than HOL
- internal *intuitionistic higher-order logic* based on *proposition-as-types* interpretation
- allows program extraction from formal proofs

Previous Work at Bell Labs by Doug Howe

- classical/computational set-theoretic semantics for Nuprl
  (proves that classical Nuprl is consistent)
- exchange format for importing external axiomatic theories into Nuprl
- Nuprl extensions for interactive instantiation of axiomatic theories
- translation from HOL theories into Nuprl theories has been implemented in ML
  (using imperative features)

Translation and Instantiation



Remaining Problems

- translator is not in a form suitable for meta-logical analysis
- correctness of the translation is crucial and should be addressed
- a general methodology to build logic translators
  with theoretical and practical guidelines is missing
- existing implementation is not a separate program
  but integrated into the HOL and Nuprl systems

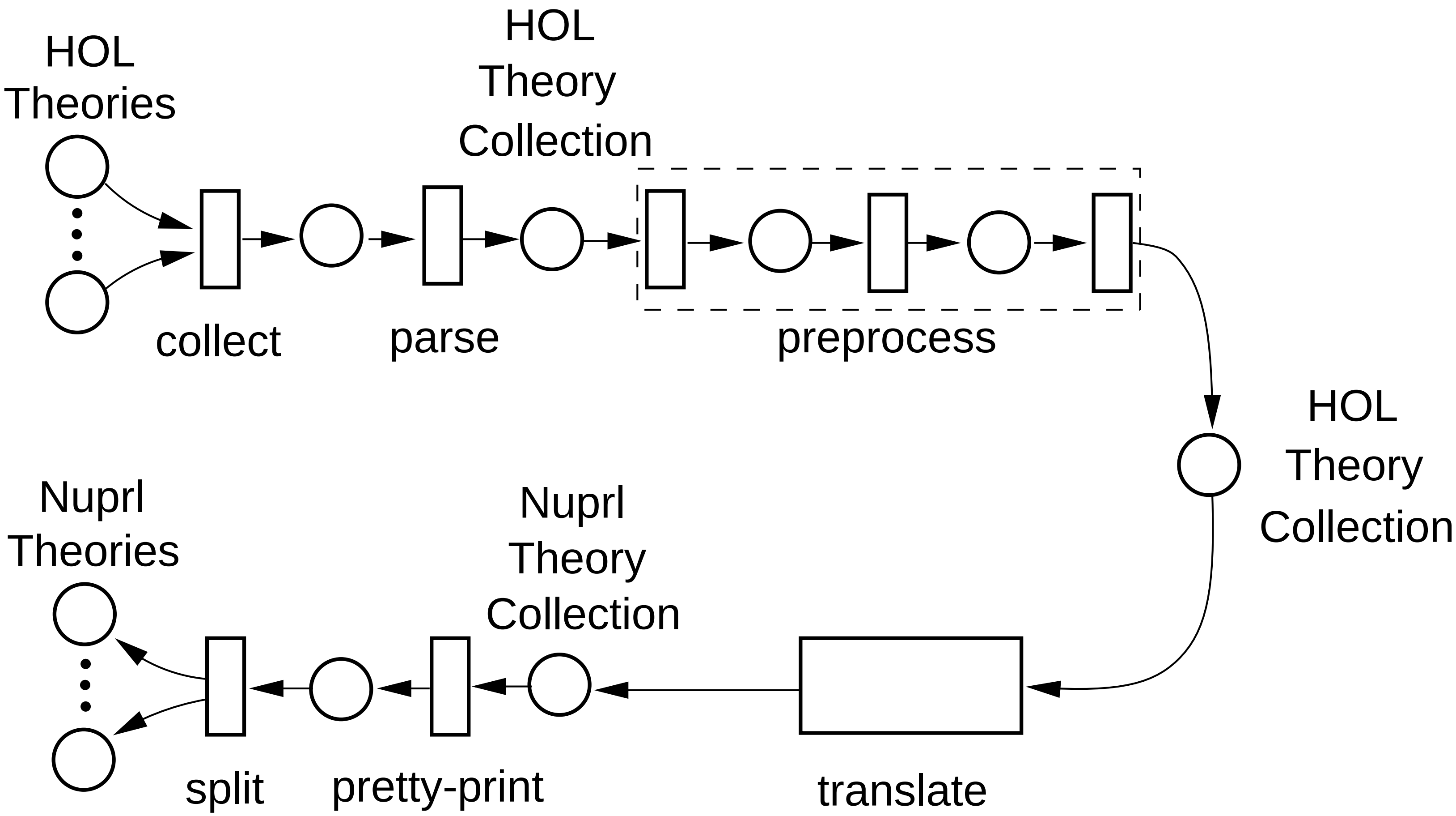
Recent Work at SRI

In order to achieve a better understanding of the translation it has been specified in Maude:

- the specification is *purely equational*,
  i.e. free of side effects and other kinds of imperativity
- the algebraic specification is more *abstract*, more *modular*
  and therefore easier to understand
- the specification is *executable*,
  hence it can be used as a running tool
- the translator has *clear external interfaces*,
  it can be understood independently of the HOL and Nuprl systems

The executable specification has been used to translate all basic HOL theories including theories of arithmetic, sets, lists, trees and a development of combinatory logic.

Architecture of the Translator



Syntactic Categories/Functors of HOL I

Objects	Category
$\Omega \ \triangleright \ \alpha^n : \mathsf{Type}^n \ \triangleright \ \sigma$	**HolFullType**
$\Omega \ \triangleright \ \Delta \ \triangleright \ \alpha^n : \mathsf{Type}^n \ \triangleright \ v^m : \sigma^m \ \triangleright \ a : \sigma$	**HolFullTerm**
$\Omega \ \triangleright \ \Delta \ \triangleright \ \alpha^n : \mathsf{Type}^n \ \triangleright \ v^m : \sigma^m \ \triangleright \ \Gamma \ \Vdash \ t$	**HolFullSen**

Objects	Functor
σ	$\mathit{HolType} : \mathbf{HolFullTypeCtxt} \to \mathbf{Set}$
$a : \sigma$	$\mathit{HolTerm} : \mathbf{HolFullCtxt} \to \mathbf{Set}$
$\alpha^n : \mathsf{Type}^n \ \triangleright \ v^m : \sigma^m \ \triangleright \ \Gamma \ \Vdash \ t$	$\mathit{HolSen} : \mathbf{HolFullSign} \to \mathbf{Cat}$

Syntactic Categories/Functors of HOL II

Objects	Category
Ω	**HolFullTypeSign = HolTypeSign**
$\Omega \triangleright \Delta$	**HolFullSign**
$\Omega \triangleright \alpha^n : \mathsf{Type}^n$	**HolFullTypeCtxt**
$\Omega \triangleright \Delta \triangleright \alpha^n : \mathsf{Type}^n \triangleright v^m : \sigma^m$	**HolFullCtxt**

Objects	Category/Functor
Ω	**HolTypeSign**
Δ	$\mathit{HolSign} : \mathbf{HolFullTypeSign} \to \mathbf{Cat}$
$\alpha^n : \mathsf{Type}^n$	$\mathit{HolTypeCtxt} : \mathbf{HolFullTypeSign} \to \mathbf{Cat}$
$v^m : \sigma^m$	$\mathit{HolCtxt} : \mathbf{HolFullTypeCtxt} \to \mathbf{Cat}$

Syntactic Categories/Functors of Nuprl I

Objects	Category
$\Sigma \ \triangleright \ v^m : \Box^m \ \triangleright \ t$	**NuprlFullTerm**
$\Sigma \ \triangleright \ v^m : S^m \ \triangleright \ t = t' \in T$	**NuprlFullSen**

Objects	Category/Functor
t	$\mathit{NuprlTerm} : \mathbf{NuprlFullCtxt} \to \mathbf{Set}$
$v^m : S^m \ \Vdash \ t = t' \in T$	$\mathit{NuprlSen} : \mathbf{NuprlFullSign} \to \mathbf{Cat}$

Syntactic Categories/Functors of Nuprl II

Objects	Category
Σ	**NuprlFullSign**
$\Sigma \triangleright v^m : \Box^m$	**NuprlFullCtxt**

Objects	Category/Functor
Σ	**NuprlSign**
$v^m : \Box^m$	**NuprlCtxt**
$v^m : S^m$	**NuprlTypedCtxt**
$v^m : S^m$	$\mathit{NuprlTypedCtxt} : \mathbf{NuprlFullSign} \to \mathbf{Cat}$
$v^m : S^m$	$\mathit{NuprlTypedCtxt} : \mathbf{NuprlFullCtxt} \to \mathbf{Cat}$

Overview of the Translation

The HOL-Nuprl translation can be specified by a functor

$$\Phi : \mathbf{HolFullSign} \to \mathbf{NuprlTh_0}$$

together with a natural transformation

$$\alpha : HolSen \to NuprlSen \circ \Phi$$

where

$$HolSen : \mathbf{HolFullSign} \to \mathbf{Cat},$$

$$NuprlSen : \mathbf{NuprlTh_0} \to \mathbf{Cat}.$$

Translating Full Signatures

First we define the functor

$$\Phi : \mathbf{HolFullSign} \to \mathbf{NurplTh_0}$$

$$\Phi(\Sigma) = (\Phi(\Sigma), \Psi(\Sigma))$$

with

$$\Phi : \mathbf{HolFullSign} \to \mathbf{NuprlFullSign}$$

$$\Psi : \prod \Sigma : \mathbf{HolFullSign} \;.\; NuprlAxSet(\Phi(\Sigma))$$

where

$$NuprlAxSet : \mathbf{NuprlFullSign} \to \mathbf{Cat}$$

Translating Type Signatures

$$\Phi : \mathbf{HolFullTypeSign} \to \mathbf{NuprlFullSign}$$

$$\Psi : \prod \Omega : \mathbf{HolFullTypeSign}\ .\ \mathit{NuprlAxSet}(\Phi(\Omega))$$

where

$$\mathit{NuprlAxSet} : \mathbf{NuprlFullSign} \to \mathbf{Cat}$$

1. $\Phi(\{\nu : \mathsf{Type}^n{\to}\mathsf{Type}\}) = \{\nu : \Box^n{\to}\Box\}$,
2. $\Phi(\emptyset) = \emptyset$,
3. $\Phi(\Omega_1 \cup \Omega_2) = \Phi(\Omega_1) \cup \Phi(\Omega_2)$
4. $\Psi(\{\nu : \mathsf{Type}^n{\to}\mathsf{Type}\}) = \{\alpha, \ldots, \alpha_n : \mathsf{S} \ \Vdash\ \nu(\alpha, \ldots, \alpha_n) \in \mathsf{S}\}$,
5. $\Psi(\emptyset) = \emptyset$,
6. $\Psi(\Omega_1 \cup \Omega_2) = \Psi(\Omega_1) \cup \Psi(\Omega_2)$

with

$$\mathsf{S} = \{T : \mathsf{U}_1 \mid \exists x : T\ .\ \mathsf{True}\},$$

Translating Types

$$\alpha : \mathit{HolType} \to \mathit{NuprlTerm} \circ \Phi$$

where

$$\mathit{HolType} : \mathbf{HolFullTypeCtxt} \to \mathbf{Set}$$

$$\Phi : \mathbf{HolFullTypeCtxt} \to \mathbf{NuprlFullCtxt}$$

$$\mathit{NuprlTerm} : \mathbf{NuprlFullCtxt} \to \mathbf{Set}$$

1. $\alpha(\Omega \, \triangleright \, \alpha^n : \mathsf{Type}^n)(\alpha_i^n) = \alpha_i^n$,

2. $\alpha(\Omega \, \triangleright \, \alpha^n : \mathsf{Type}^n)(\mathsf{o}) = \mathsf{Bool}$,

3. $\alpha(\Omega \, \triangleright \, \alpha^n : \mathsf{Type}^n)(\sigma \to \tau) =$
   $\alpha(\Omega \, \triangleright \, \alpha^n : \mathsf{Type}^n)(\sigma) \to \alpha(\Omega \, \triangleright \, \alpha^n : \mathsf{Type}^n)(\tau)$,

4. $\alpha(\Omega \, \triangleright \, \alpha^n : \mathsf{Type}^n)(\nu(\sigma_1, \ldots, \sigma_n)) =$
   $\nu(\alpha(\Omega \, \triangleright \, \alpha^n : \mathsf{Type}^n)(\sigma_1), \ldots, \alpha(\Omega \, \triangleright \, \alpha^n : \mathsf{Type}^n)(\sigma_n))$

Translating Signatures

$$\alpha : \mathit{HolSign} \to \mathbf{NuprlSign}$$

$$\beta(\Omega) : \prod \Delta : \mathit{HolSign}(\Omega) \,.\, \mathit{NuprlAxSet}(\alpha(\Omega)(\Delta))$$

where

$$\mathit{HolSign} : \mathbf{HolTypeSign} \to \mathbf{Cat}$$

$$\mathbf{NuprlSign} : \mathbf{HolTypeSign} \to \mathbf{Cat}$$

$$\mathit{NuprlAxSet} : \mathbf{NuprlFullSign} \to \mathbf{Cat}$$

1. $\alpha(\Omega)(\{\alpha^n : \mathsf{Type}^n \;\triangleright\; c : \sigma\}) = \{c : \Box^n \to \Box\}$,
2. $\alpha(\Omega)(\emptyset) = \emptyset$,
3. $\alpha(\Omega)(\Delta_1 \cup \Delta_2) = \alpha(\Omega)(\Delta_1) \cup \alpha(\Omega)(\Delta_2)$
4. $\beta(\Omega)(\{\alpha^n : \mathsf{Type}^n \;\triangleright\; c : \sigma\}) =$
   $\{\alpha^n : \mathsf{S}^n \;\Vdash\; c(\alpha^n) \in \alpha(\Omega \;\triangleright\; \alpha^n : \mathsf{Type}^n)(\sigma)\}$,
5. $\beta(\Omega)(\emptyset) = \emptyset$,
6. $\beta(\Omega)(\Delta_1 \cup \Delta_2) = \beta(\Omega)(\Delta_1) \cup \beta(\Omega)(\Delta_2)$

Translating Terms

$$\alpha : \mathit{HolTerm} \to \mathit{NuprlTerm} \circ \Phi$$

where

$$\mathit{HolTerm} : \mathbf{HolFullCtxt} \to \mathbf{Set}$$

$$\Phi : \mathbf{HolFullCtxt} \to \mathbf{NuprlFullCtxt}$$

$$\mathit{NuprlTerm} : \mathbf{NuprlFullCtxt} \to \mathbf{Set}$$

1. $\alpha(\Omega \triangleright \Delta \triangleright \alpha^n : \mathsf{Type}^n \triangleright v^m : \sigma^m)(v_i^m) = v_i^m$,

2. $\alpha(\Omega \triangleright \Delta \triangleright \alpha^n : \mathsf{Type}^n \triangleright v^m : \sigma^m)(= : \tau \to \tau \to \mathsf{o}) = (=_b)(\alpha(\Omega)(\tau))$,

3. $\alpha(\Omega \triangleright \Delta \triangleright \alpha^n : \mathsf{Type}^n \triangleright v^m : \sigma^m)(\Rightarrow : \mathsf{o} \to \mathsf{o} \to \mathsf{o}) = (\Rightarrow_b)$,

4. $\alpha(\Omega \triangleright \Delta \triangleright \alpha^n : \mathsf{Type}^n \triangleright v^m : \sigma^m)(c : \tau[\tau^k / \beta^k]) =$
   $c(\alpha(\Omega \triangleright \alpha^n : \mathsf{Type}^n)(\tau_1^n), \ldots, \alpha(\Omega \triangleright \alpha^n : \mathsf{Type}^n)(\tau_n^n))$,
   where $(\beta^k : \mathsf{Type}^k \triangleright c : \tau) \in \Delta$ and we have types $\alpha^n : \mathsf{Type}^n \triangleright \tau_i^n$

5. $\alpha(\Omega \triangleright \Delta \triangleright \alpha^n : \mathsf{Type}^n \triangleright v^m : \sigma^m)(((a : \sigma \to \tau)(b : \sigma)) : \tau) =$
   $(\alpha(\Omega \triangleright \Delta \triangleright \alpha^n : \mathsf{Type}^n \triangleright v^m : \sigma^m)(a : \sigma \to \tau)$
   $\alpha(\Omega \triangleright \Delta \triangleright \alpha^n : \mathsf{Type}^n \triangleright v^m : \sigma^m)(b : \sigma))$,

6. $\alpha(\Omega \triangleright \Delta \triangleright \alpha^n : \mathsf{Type}^n \triangleright v^m : \sigma^m)((\lambda\, v : \sigma \,.\, (a : \tau)) : \sigma \to \tau) =$
   $(\lambda\, v \,.\, \alpha(\Omega \triangleright \Delta \triangleright \alpha^n : \mathsf{Type}^n \triangleright v^m : \sigma^m v : \sigma)(a : \tau))$

with

- $\Rightarrow_b = \lambda\ x, y$. if x then y else tt,
- $=_b(\rho) = \lambda\ x, y\ .\ \downarrow_b (x = y \in \rho)$.

Translating Sentences

$$\alpha : HolSen \to NuprlSen \circ \Phi$$

where

$$HolSen : \mathbf{HolFullSign} \to \mathbf{Cat}$$

$$NuprlSen : \mathbf{NuprlFullSign} \to \mathbf{Cat}$$

$$\Phi : \mathbf{HolFullSign} \to \mathbf{NuprlFullSign}$$

$$\alpha(\Omega \; \triangleright \; \Delta)(\alpha^n : \mathsf{Type}^n \; \triangleright \; v^m : \sigma^m \; \triangleright \; \Gamma \; \Vdash \; t) =$$

$$\Phi(\alpha^n : \mathsf{Type}^n)$$

$$\alpha(\Omega \; \triangleright \; \alpha^n : \mathsf{Type}^n)(v^m : \sigma^m)$$

$$\triangleright$$

$$\alpha(\Omega \; \triangleright \; \Delta \; \triangleright \; \alpha^n : \mathsf{Type}^n \; \triangleright \; v^m : \sigma^m)(\Gamma)$$

$$\Vdash$$

$$\mathsf{axiom} \in \uparrow \alpha(\Omega \; \triangleright \; \Delta \; \triangleright \; \alpha^n : \mathsf{Type}^n \; \triangleright \; v^m : \sigma^m)(t)$$

with

1. $\alpha(\Omega \,\triangleright\, \Delta \,\triangleright\, \alpha^n : \mathsf{Type}^n \,\triangleright\, v^m : \sigma^m)([]) = []$

2. $\alpha(\Omega \,\triangleright\, \Delta \,\triangleright\, \alpha^n : \mathsf{Type}^n \,\triangleright\, v^m : \sigma^m)([t_1, \ldots, t_n]) =$
   $[\ h_1\ :\uparrow \alpha(\Omega \,\triangleright\, \Delta \,\triangleright\, \alpha^n : \mathsf{Type}^n \,\triangleright\, v^m : \sigma^m)(t_1), \ldots,$
   $h_n\ :\uparrow \alpha(\Omega \,\triangleright\, \Delta \,\triangleright\, \alpha^n : \mathsf{Type}^n \,\triangleright\, v^m : \sigma^m)(t_n)\ ]$

Correctness of the Translation

The HOL-Nuprl translation satisfies a syntactical correctness condition, namely that it constitutes a

map between entailment systems

of HOL and Nuprl in the sense of general logics.

Apart from categorical well-formedness the essential condition to be verified is

$$\Gamma \vdash_{Hol} \phi \text{ implies } \alpha_\Sigma(\Gamma) \vdash_{ClassicalNuprl} \alpha_\Sigma(\phi)$$
$$\text{for each full HOL signature } \Sigma$$

$\Gamma \vdash \phi$ means that the sentence ϕ can be *derived (by a sequential proof)* using the sentences in Γ as hypotheses.

Sequential proofs are defined by induction using *instances of inference rules*.

Hence to verify the condition above it is sufficient to show that each translated instance of an HOL-inference rule can be derived in Classical Nuprl.

HOL-Rules I

$$\frac{}{\alpha^n : \mathsf{Type}^n \ \triangleright\ v^m : \sigma^m \ \triangleright\ t \ \Vdash\ t} \qquad \text{(ASSUME)}$$

$$\frac{\alpha^n : \mathsf{Type}^n \ \triangleright\ v^m : \sigma^m \ \triangleright\ H \ \Vdash\ t}{\alpha^n : \mathsf{Type}^n \ \triangleright\ v^m : \sigma^m \ \triangleright\ H' \ \Vdash\ t} \ \text{if } H \text{ and } H' \text{ are equal as sets} \qquad \text{(STRUCT)}$$

$$\frac{\alpha^n : \mathsf{Type}^n \ \triangleright\ v^m : \sigma^m \ \triangleright\ H_2 \ \Vdash\ t_1 \qquad \alpha^n : \mathsf{Type}^n \ \triangleright\ v^m : \sigma^m \ \triangleright\ H_1 \ \Vdash\ t_1 \Rightarrow t_2}{\alpha^n : \mathsf{Type}^n \ \triangleright\ v^m : \sigma^m \ \triangleright\ H_1, H_2 \ \Vdash\ t_2} \qquad \text{(MP)}$$

$$\frac{\alpha^n : \mathsf{Type}^n \ \triangleright\ v^m : \sigma^m \ \triangleright\ H \ \Vdash\ t_2}{\alpha^n : \mathsf{Type}^n \ \triangleright\ v^m : \sigma^m \ \triangleright\ H - \{t_1\} \ \Vdash\ t_1 \Rightarrow t_2} \qquad \text{(DISCH)}$$

$$\frac{}{\alpha^n : \mathsf{Type}^n \ \triangleright\ v^m : \sigma^m \triangleright\ \Vdash\ t = t} \qquad \text{(REFL)}$$

HOL-Rules II

$$\frac{}{\alpha^n : \mathsf{Type}^n \;\triangleright\; v^m : \sigma^m \triangleright \;\Vdash\; ((\lambda\, v : \sigma\, .\, s)\, t) = s[t/v]} \quad \text{(BETA\_CONV)}$$

$$\frac{\begin{array}{c}\alpha^n : \mathsf{Type}^n \;\triangleright\; v^m : \sigma^m \;\triangleright\; H_i^k \;\Vdash\; t_i^k = t_i'^k \text{ for all } i \\ \alpha^n : \mathsf{Type}^n \;\triangleright\; v^m : \sigma^m \;\triangleright\; H \;\Vdash\; t[t^k/z]\end{array}}{\alpha^n : \mathsf{Type}^n \;\triangleright\; v^m : \sigma^m \;\triangleright\; H^k, H \;\Vdash\; t[t'^k/z]} \quad \text{(SUBST)}$$

$$\frac{\alpha^n : \mathsf{Type}^n \;\triangleright\; v^m : \sigma^m \;\triangleright\; H \;\Vdash\; s = t}{\alpha^n : \mathsf{Type}^n \;\triangleright\; v^m : \sigma^m \;\triangleright\; H \;\Vdash\; (\lambda\, v : \sigma\, .\, s) = (\lambda\, v : \sigma\, .\, t)} \quad \text{(ABS)}$$

$$\frac{\alpha^n : \mathsf{Type}^n \;\triangleright\; v^m : \sigma^m \;\triangleright\; H \;\Vdash\; t}{\beta^k : \mathsf{Type}^k \;\triangleright\; (v^m : \sigma^m \;\triangleright\; H \;\Vdash\; t)[\tau^n/\alpha^n]} \quad \text{(INST\_TYPE)}$$

provided that we have types $\beta^k : \mathsf{Type}^k \;\triangleright\; \tau_i^n$ and the α_i^n do not occur in H.

Remaining Questions/Future Work

- What alternative translations are conceivable and
  how can they explained in the framework of general logics ?

- What are the right categorical methods/tools
  to deal with logics and type theories on a syntactical level ?

- What is the role of meta logics like ISABELLE, MEL, MetaPRL in this context ?

- Is there a need/use of a semantic connection (e.g. in the sense of institutions) ?

- Is there a categorical simplification/generalization of Allen's Semantics ?

- Explore other connections within the family of
  higher-order logic theorem provers (Nuprl, HOL, PVS, COQ, LEGO)

- Apply the methodology to develop further translators
  which can be directly used as valuable tools by formal method researchers.