Proof-theoretic Conservative Extension for HOL with Ad-hoc Overloading

Arve Gengelbach   Tjark Weber

Uppsala University, Uppsala, Sweden

ICTAC, December 2, 2020
Motivation

What’s in a definition?
That is, what new theorems can be derived with a definition?

Sufficient criterion for when a formula
is also provable without some definitions
Motivation (Example)

Declared constant: $d_\alpha$
$T = \{d_\alpha \text{ list} \equiv \ldots\}$

What new theorems can be derived with a definition of $d_{\text{bool}}$?

Assume $d_{\text{bool}}$ not occurring in the definition of $d_\alpha \text{ list}$.
Any formula without $d_{\text{bool}}$ is provable from $T$ iff the formula is provable from $T \cup \{d_{\text{bool}} \equiv \text{True}\}$.
$T \cup \{d_{\text{bool}} \equiv \text{False}\}$. 

Higher-Order Logic (HOL)

- Typed λ-calculus with rank 1 polymorphism
- With built-in types →, bool
  and a built-in constant =_α→α→bool
- Theories of type and constant definitions
- Constants may be overloaded
  Example: +_α→α→α may be overloaded at
  +_real→real→real and +_nat→nat→nat
Definition: Proof-theoretic Conservativity

Theory extension by definition(s) is \textit{proof-theoretically conservative} if for all theories $T \subseteq T'$ and for all formulae $\varphi$ (in the language of $T$) we have:

$$T' \vdash \varphi \iff T \vdash \varphi$$

Initial Example:
$T = \{d_{\alpha} \text{ list} \equiv \ldots\}$ and $T' = T \cup \{d_{\text{bool}} \equiv \text{True}\}$ have the same languages.

Problem:
$T' \vdash d_{\text{bool}} = \text{True}$ and $T \not\vdash d_{\text{bool}} = \text{True}$. 
In HOL with ad-hoc overloading, definitions are conservative in the following sense:

If a formula $\varphi$ is independent\(^1\) of symbols defined in $T' \setminus T$ then

$$T' \vdash \varphi \iff T \vdash \varphi$$

\(^1\)expressed by closure of definitional dependencies
Example:
Proof-theoretic Conservativeness done right

\[ T = \{ d_\alpha \text{list} \equiv \ldots \} \text{ and } T' = T \cup \{ d_{\text{bool}} \equiv \text{True} \}. \]
Assume \( d_{\text{bool}} \) not occurring in the definition of \( d_\alpha \text{list} \).

- Any formula \( \varphi \) without \( d_{\text{bool}} \) is independent of \( d_{\text{bool}} \),
  thus \( T' \vdash \varphi \iff T \vdash \varphi \)
- \( T' \vdash c_\alpha \text{list} = d_\alpha \text{list} \iff T \vdash c_\alpha \text{list} = d_\alpha \text{list} \)
- But we can **not** prove
  \( T' \vdash c_{\text{bool}} = d_{\text{bool}} \iff T \vdash c_{\text{bool}} = d_{\text{bool}} \)
Generalise semantics
Relaxed interpretation of function types: \([\tau \to \sigma] \subseteq [\tau] \to [\sigma]\)
Only interpret type-variable free constants and types.

HOL with ad-hoc overloading is sound and complete
(Andrews/Henkin)

Prove model-theoretic conservativity to obtain proof-theoretic conservativity for theories of definitions
Foundation of Isabelle/HOL:
Consistency; Definitions are abbreviations
Practical: Ignore unrelated definitions in proof-search
Any theory of definitions $T'$ is a proof-theoretically conservative extension of the initial theory.

Meta-safety: Definitions in a formula can be unfolded, resulting in a logically equivalent formula.

We can recover this result (not meta-safety).
What’s in a definition?

All the definitions that are neither implicitly nor explicitly relevant to a formula are irrelevant to the provability of the formula.
Track a definition’s dependencies (to disallow cyclic definitions).

- $u \equiv t$, and $v \in t^* \cup t^\circ$ then $u \rightsquigarrow v$
  
  Example: $2\mathbb{N} \rightsquigarrow \mathbb{N}$, \hspace{1em} $2\mathbb{N} \rightsquigarrow \text{even}_{\mathbb{N} \rightarrow \text{bool}}$
  
  (from a definition $2\mathbb{N} \equiv \text{even}_{\mathbb{N} \rightarrow \text{bool}}$)

- $c_\sigma \rightsquigarrow v$ for $v \in \sigma^*$
  
  Example: $\text{map}(\alpha \rightarrow \beta) \rightarrow \alpha \text{ list} \rightarrow \beta \text{ list} \rightsquigarrow v$
  
  for $v \in \{\alpha, \beta, \alpha \text{ list}, \beta \text{ list}\}$

- $(\sigma_1, \ldots, \sigma_n) k \rightsquigarrow \sigma_i$ for a type constructor $k$
  
  Reason/Example: $\tau \rightarrow \sigma \rightsquigarrow \tau$ \hspace{1em} and \hspace{1em} $\tau \rightarrow \sigma \rightsquigarrow \sigma$
The $U$-independent fragment$^2$ is

\[ F_U := \text{Symb} \setminus \{x \mid \exists u \in U, \rho. \ x \xrightarrow{\cdots} \rho(u)\}. \]

$F_U$ contains all symbols that are not depending on any instance from $U$.

Typically, $U = \text{set of symbols defined by definitions}$.

**Example:** $c_\alpha \equiv d_\alpha$, $d_{\text{bool}} \equiv \text{True}$

\[ c_{\text{bool}}, d_{\text{bool}} \not\in F\{d_{\text{bool}}\} \quad \text{by} \quad c_{\text{bool}} \xrightarrow{\cdots} \rho(u) \]

\[ c_\alpha \text{ list} \in F\{d_{\text{bool}}\} \quad \text{by} \quad c_\alpha \text{ list} \xrightarrow{\cdots} \rho(u) \]

\[ ^2 \quad \downarrow \text{type-substitutive closure}, \quad \ast \text{ reflexive-transitive closure} \]
Let $T \subseteq T'$ be a wellformed definitional extension and let $\mathcal{M}$ be a model of $T$. There exists a model $\mathcal{M}'$ of $T'$ such that $\mathcal{M}$ and $\mathcal{M}'$ interpret terms equally, that are independent of symbols defined in $T' \setminus T$.

Example: $T = \{ c_\alpha \equiv d_\alpha \}$, $T' = T \cup \{ d_{\text{bool}} \equiv \text{True} \}$. Any model $\mathcal{M}$ for $T$ has an extension $\mathcal{M}'$ for $T'$.

$c_{\text{bool}}$ is dependent on $d_{\text{bool}}$, thus may have $\mathcal{M}(c_{\text{bool}}) \neq \mathcal{M}'(c_{\text{bool}})$. But $\mathcal{M}(c_\alpha \text{ list}) = \mathcal{M}'(c_\alpha \text{ list})$. 
Model-theoretic Conservativity implies Proof-theoretic Conservativity

Theorem: Let $T \subseteq T'$ be a wellformed definitional extension. If $\varphi$ independent from the symbols defined in $T' \setminus T$ and $T' \vdash \varphi$ then $T \vdash \varphi$.

Proof:
With completeness it suffices to prove: $\varphi$ holds in all models of $T$. For a model $M$ of $T$ model-theoretic conservativity gives a model $M'$ of $T'$ such that $M(\varphi) = M'(\varphi)$. From $T' \vdash \varphi$ soundness gives $M'(\varphi) = \text{true}$, thus $M(\varphi) = \text{true}$. 
Lazy Ground Semantics [Åman Pohjola & Gengelbach 2020]

- \( \varphi_{\text{bool}} \) satisfied w.r.t. \([\cdot]\) iff
  
  for all ground type substitutions \( \rho \)
  and all variable assignments \( \xi_\rho : [\varphi]_{\xi_\rho} = \text{true} \)

- Earlier semantics: \( [\rho(\varphi)]_{\xi_\rho} = \text{true} \)
  Problem: Term variables \( x_\alpha \) and \( x_{\text{bool}} \) are distinct,
          but immediately applying \( \rho \) equates these.

- Lazy semantics
  
  For \( \rho(\sigma) \) ground type, have \( [c_\sigma]_{\xi_\rho} = [c_{\rho(\sigma)}] \) and
  \( [x_\sigma]_{\xi_\rho} = \xi_\rho(x_\sigma) \) such that \( \xi_\rho(x_\sigma) \in [\rho(\sigma)] \)

- \( \mathcal{M} \models T \) iff every \( \varphi_{\text{bool}} \in T \) is satisfied w.r.t. \( \mathcal{M} \).