Automating Quantified Conditional Logics in HOL*

Christoph Benzmüller
Dep. of Mathematics and Computer Science
Freie Universität Berlin, Germany
c.benzmueller@fu-berlin.de
http://christoph-benzmueller.de

Abstract

A notion of quantified conditional logics is provided that includes quantification over individual and propositional variables. The former is supported with respect to constant and variable domain semantics. In addition, a sound and complete embedding of this framework in classical higher-order logic is presented. Using prominent examples from the literature it is demonstrated how this embedding enables effective automation of reasoning within (object-level) and about (meta-level) quantified conditional logics with off-the-shelf higher-order theorem provers and model finders.

1 Introduction

Conditional logics (CL), known also as logics of normality or typicality, have many applications including action planning, counterfactual reasoning, default reasoning, deontic reasoning, metaphysical modeling and reasoning about knowledge. While there is broad literature on propositional CLs only a few authors have addressed first-order extensions, those include Delgrande [1998] and Friedman et al. [2000].

This paper presents (in Sec. 2)—as first contribution—a notion of quantified conditional logics (QCLs) that includes quantification over propositional and first-order variables, a combination that has not been addressed yet. An adequate semantics of conditionality ($\Rightarrow$) is achieved by adapting selection function semantics [Stalnaker, 1968; Chellas, 1975]. Regarding quantification over individuals constant domains and varying domains are supported. The treatment of propositional quantification is inspired by Fitting's [2002] work on quantified modal logics. Employing propositional quantification is particularly useful for flexible automation of object- and meta-level reasoning. For example, 'syntactic' conditional logic axioms such as $\forall P(P \Rightarrow P)$ (cf. Ex. 2) can simply be postulated and even interesting meta-properties, including correspondence and dependence/independence results for such axioms can be automatically established. When combining logics, bridge principles play a crucial role. Using propositional quantification such bridge principles can again be simply postulated as axioms. In traditional systems they are often most challenging to implement.

A semantic embedding of propositional CLs in classical higher-order logic HOL (Church's [1940] type theory; cf. Sec. 3) has been presented by Benzmüller et al. [2012a]. This embedding exploits the natural correspondence between selection function semantics and HOL. In fact, selection function semantics can be seen as an higher-order extension of well-known Kripke semantics for modal logic and cannot be naturally embedded into first-order logic.

This paper extends—as second contribution—the embedding of Benzmüller et al. [2012a] to also include quantification over propositional and individual variables, the latter w.r.t. constant nd varying domains (Sec. 4). This embedding of QCL in HOL is shown sound and complete (in Sec. 5).

So far, the work on first-order CLs has focused mainly on theory and practically available prover implementations do not exist. Due to the timeliness and the broad range of CL applications, implementations and evaluations of respective provers should be strongly fostered though.

Therefore this paper offers—as third contribution—an elegant solution to automate QCLs with off-the-shelf theorem provers and model finders for HOL. A unique feature of the embeddings-based approach is that it supports both reasoning at the object-level and at the meta-level. This will be illustrated with prominent examples from the literature (in Sec. 6).

A discussion of related and future work concludes the paper (in Secs. 7 and 8).

2 Quantified Conditional Logics

Propositional CLs are extended with quantification over *propositional variables* and over *individual variables*. Regarding the latter *constant domains* (every possible world has the same domain) and *varying domains* (different possible worlds may have different domains) are supported; in this regard the framework below is related to that of Delgrande [1998]. However, the inclusion of propositional quantification is novel. The gained expressivity is of crucial significance for the automation of QCLs with HOL provers and model finders (cf. Sec. 6).

DEF. 2.1 (QCL) *Let IV be a set of first-order (individual) variables, PV a set of propositional variables, and SYM a set of predicate symbols of any arity.* Formulas *of QCL are given*

[*] This work has been supported by the German Research Foundation under grant BE2501/9-1.

by the following grammar (where $X^i \in IV, P \in PV, k \in$ SYM, and where $\Rightarrow$ represents conditionality):

$$\begin{aligned}\varphi,\psi ::= \;& P \mid k(X^1,\ldots,X^n) \mid \neg\varphi \mid \varphi\vee\psi \mid \varphi\Rightarrow\psi \mid \\ & \forall^{co}X\varphi \mid \forall^{va}X\varphi \mid \forall P\varphi\end{aligned}$$

From this set of primitive connectives*, other logical connectives can be introduced as abbreviations: for example, $\varphi\wedge\psi$, $\varphi\rightarrow\psi$ (material implication), $\varphi\leftrightarrow\psi$ and $\Box\varphi$ abbreviate $\neg(\neg\varphi\vee\neg\psi)$, $\neg\varphi\vee\psi$, $(\varphi\rightarrow\psi)\wedge(\psi\rightarrow\varphi)$ and $\neg\varphi\Rightarrow\varphi$, respectively. $\forall^{co}$ and $\forall^{va}$ are associated with constant domain and variable domain quantification. For $* \in \{co, va\}$, $\exists^{*}X\varphi$ abbreviates $\neg\forall^{*}X\neg\varphi$. Syntactically, QCL can be seen as a generalization of quantified multimodal logic where the index of modality $\Rightarrow$ is a formula of the same language. For instance, in $(\varphi\Rightarrow\psi)\Rightarrow\delta$ the subformula $\varphi\Rightarrow\psi$ is the index of the second occurrence of $\Rightarrow$.*

Regarding semantics, different formalizations have been proposed (see [Nute, 1980]). The work presented here builds on *selection function semantics* [Stalnaker, 1968; Chellas, 1975], which is based on possible world structures and has been successfully used by Olivetti et al. [2007] to develop proof methods for some propositional CLs.

DEF. 2.2 (Interpretation) *An* interpretation *is a structure $M = \langle S, f, D, D', Q, I\rangle$ where, S is a set of possible items called worlds, $f : S \times 2^S \mapsto 2^S$ is the selection function, D is a non-empty set of* individuals *(the constant first-order domain), D' is a function that assigns a non-empty subset $D'(w)$ of D to each possible world w (the $D'(w)$ are the varying domains), Q is a non-empty collection of subsets of W (the propositional domain), and I is a classical interpretation function where for each n-ary predicate symbol k, $I(k,w) \subseteq D^n$.*

DEF. 2.3 (Assignment) *A* variable assignment $g = (g^i, g^p)$ *is a pair of maps where, $g^i : IV \mapsto D$ maps each individual variable in IV to an object in D, and $g^p : PV \mapsto Q$ maps each propositional variable in PV to a set of worlds in Q.*

DEF. 2.4 (Satisfiability) Satisfiability *of a formula φ for an interpretation $M = \langle S, f, D, D', Q, I\rangle$, a world $s \in S$, and a variable assignment $g = (g^i, g^p)$ is denoted as $M, g, s \models \varphi$ and defined as follows, where $[a/Z]g$ denotes the assignment identical to g except that $([a/Z]g)(Z) = a$:*

$M, g, s \models k(X^{1}_{,}\ldots, X^n)$ *iff* $\langle g^i(X^1),\ldots, g^i(X^n)\rangle \in I(k,w)$

$M, g, s \models P$ *iff* $s \in g^p(P)$

$M, g, s \models \neg\varphi$ *iff* $M, g, s \not\models \varphi$ *(that is, not* $M, g, s \models \varphi$*)*

$M, g, s \models \varphi\vee\psi$ *iff* $M, g, s \models \varphi$ *or* $M, g, s \models \psi$

$M, g, s \models \varphi\Rightarrow\psi$ *iff* $\forall t \in S$ *with* $t \in f(s, [\varphi])$ *(where*
$[\varphi] := \{u \mid M, g, u \models \varphi\}$*) holds:* $M, g, t \models \psi$

$M, g, s \models \forall^{co}X\varphi$ *iff* $\forall d \in D$: $M, ([d/X]g^i, g^p), s \models \varphi$

$M, g, s \models \forall^{va}X\varphi$ *iff* $\forall d \in D'(s)$: $M, ([d/X]g^i, g^p), s \models \varphi$

$M, g, s \models \forall P\varphi$ *iff* $\forall p \in Q$: $M, (g^i, [p/P]g^p), s \models \varphi$

DEF. 2.5 (Model, validity) *An interpretation $M = \langle S, f, D, D', Q, I\rangle$ is a QCL* model *if for every variable assignment g and every formula φ, the set of worlds $\{s \in S \mid M, g, s \models \varphi\}$ is a member of Q. (This requirement, which is inspired by Fitting [2002], Def. 3.5, ensures a natural correspondence to Henkin models in HOL; cf. Sec. 5 and Footnote 4; cf. also Benzmüller and Paulson [2013].) As usual, a conditional formula φ is* valid in a QCL model $M = \langle S, f, D, D', Q, I\rangle$, *denoted with $M \models \varphi$, iff for all worlds $s \in S$ and variable assignments g holds $M, g, s \models \varphi$. A formula φ is a* valid, *denoted $\models \varphi$, iff it is valid in every QCL model.*

f is defined to take $[\varphi]$ (called the *proof set* of φ w.r.t. a given QCL model M) instead of φ. This approach has the consequence of forcing the so-called *normality* property: given a QCL model M, if φ and φ' are equivalent (i.e., they are satisfied in the same set of worlds), then they index the same formulas w.r.t. to the $\Rightarrow$ modality. The axiomatic counterpart of the normality condition is given by the rule (RCEA)

$$\frac{\varphi\leftrightarrow\varphi'}{(\varphi\Rightarrow\psi)\leftrightarrow(\varphi'\Rightarrow\psi)}\,(RCEA)$$

Moreover, it can be easily shown that the above semantics forces also the following rules to hold:

$$\frac{(\varphi_1\wedge\ldots\wedge\varphi_n)\leftrightarrow\psi}{(\varphi_0\Rightarrow\varphi_1\wedge\ldots\wedge\varphi_0\Rightarrow\varphi_n)\rightarrow(\varphi_0\Rightarrow\psi)}\,(RCK)$$

$$\frac{\varphi\leftrightarrow\varphi'}{(\psi\Rightarrow\varphi)\leftrightarrow(\psi\Rightarrow\varphi')}\,(RCEC)$$

QCK (cf. CK in [Chellas, 1980]) denotes the minimal QCL closed under rules RCEA, RCEC and RCK. In what follows, only QCLs extending QCK are considered.

3 Classical Higher-Order Logic (HOL)

HOL is a logic based on simply typed λ-calculus [Church, 1940; Andrews, 2009]. The set T of simple types in HOL is usually freely generated from a set of basic types $\{o, i\}$ using the function type constructor $\rightarrow$. Here a set of basic types $\{o, i, u\}$ is considered instead, where o denotes the type of Booleans, and where i and u denote some non-empty domains. Without loss of generality, i will later be associated with a set of worlds and u with a domain of individuals.

DEF. 3.1 (HOL) *Let $\alpha, \beta, o \in T$. The* terms *of* HOL *are defined by the grammar (c_α denotes typed constants and X_α typed variables distinct from c_α):*

$$\begin{aligned}s,t ::= \;& c_\alpha \mid X_\alpha \mid (\lambda X_\alpha s_\beta)_{\alpha\rightarrow\beta} \mid (s_{\alpha\rightarrow\beta}\, t_\alpha)_\beta \mid (\neg_{o\rightarrow o}\, s_o)_o \mid \\ & (s_o \vee_{o\rightarrow o\rightarrow o} t_o)_o \mid (\Pi_{(\alpha\rightarrow o)\rightarrow o}\, s_{\alpha\rightarrow o})_o\end{aligned}$$

Complex typed terms are constructed via abstraction and application. The primitive logical connectives are $\neg_{o\rightarrow o}, \vee_{o\rightarrow o\rightarrow o}$ and $\Pi_{(\alpha\rightarrow o)\rightarrow o}$ (for each type α). From these, other logical connectives can be introduced as abbreviations: for example, $\wedge$ and $\rightarrow$ abbreviate the terms $\lambda A\lambda B\,\neg(\neg A\vee\neg B)$ and $\lambda A\lambda B\,\neg A\vee B$, etc. HOL terms of type o are called formulas. Binder notation $\forall X_\alpha\, s_o$ is used as an abbreviation for $\Pi_{(\alpha\rightarrow o)\rightarrow o}\lambda X_\alpha s_o$. Type information as well as brackets may be omitted in the remainder if obvious from the context. Substitution of a term A_α for a variable X_α in a term B_β is denoted by $[A/X]B$, where it is assumed that the bound *variables of B avoid variable capture. Well known operations and relations on HOL terms include $\beta\eta$-normalization and $\beta\eta$-equality, denoted by $s =_{\beta\eta} t$.*

The following definition of HOL semantics closely follows the standard literature [Andrews, 2009; 1972a; 1972b].

DEF. 3.2 (Frame) *A* frame *is a collection $\{D_\alpha\}_{\alpha \in T}$ of nonempty sets called* domains *such that $D_o = \{T, F\}$ where T represents truth and F falsehood, $D_i \neq \emptyset$ and $D_u \neq \emptyset$ are chosen arbitrary, and $D_{\alpha\rightarrow\beta}$ are collections of total functions mapping D_α into D_β.*

DEF. 3.3 (Interpretation) *An* interpretation *is a tuple $\langle\{D_\alpha\}_{\alpha\in T}, I\rangle$ where $\{D_\alpha\}_{\alpha\in T}$ is a frame and where function I maps each typed constant symbol c_α to an appropriate element of D_α, which is called the* denotation *of c_α. The denotations of $\neg, \vee$ and $\Pi_{(\alpha\rightarrow o)\rightarrow o}$ are always chosen as usual. (Moreover, the $D_{\alpha\rightarrow\alpha\rightarrow o}$ are required to contain the respective identity relations [Andrews, 1972b].)*

DEF. 3.4 (Variable Assignment) *A* variable assignment ϕ *maps variables X_α to elements in D_α.*

DEF. 3.5 (Henkin model) *An interpretation is a* Henkin model (general model) *iff there is a binary valuation function V such that $V(\phi, s_\alpha) \in D_\alpha$ for each variable assignment ϕ and term s_α, and the following conditions are satisfied for all ϕ, variables X_α, constants c_α, and terms $l_{\alpha\rightarrow\beta}, r_\alpha, s_\beta$ (for $\alpha, \beta \in T$): $V(\phi, X_\alpha) = \phi(X_\alpha)$, $V(\phi, c_\alpha) = I(c_\alpha)$, $V(\phi, l_{\alpha\rightarrow\beta}\, r_\alpha) = (V(\phi, l_{\alpha\rightarrow\beta})V(\phi, r_\alpha))$, and $V(\phi, \lambda X_\alpha s_\beta)$ represents the function from D_α into D_β whose value for each argument $z \in D_\alpha$ is $V(\phi[z/X_\alpha], s_\beta)$, where $\phi[z/X_\alpha]$ is that assignment such that $\phi[z/X_\alpha](X_\alpha) = z$ and $\phi[z/X_\alpha]Y_\beta = \phi Y_\beta$ when $Y_\beta \neq X_\alpha$.*

PROP. 3.6 *If an interpretation $H = \langle\{D_\alpha\}_{\alpha\in T}, I\rangle$ is an Henkin model the function V is uniquely determined and $V(\phi, s_\alpha) \in D_\alpha$ is called the* denotation *of s_α.*

DEF. 3.7 (Standard model) *H is called a* standard model *iff for all α and β, $D_{\alpha\rightarrow\beta}$ is the set of all functions from D_α into D_β. It is easy to verify that each standard model is also a Henkin model.*

DEF. 3.8 (Validity) *A formula A of HOL is* valid *in a Henkin model H iff $V(\phi, A) = T$ for all variable assignments ϕ. In this case the notation $H \models A$ is used. A is (Henkin) valid, denoted as $\models A$, iff $H \models A$ for all Henkin models H. If S is a set of formulas, then $S \models A$ expresses that $H \models A$ holds for all Henkin models H with $H \models B$ for all $B \in S$.*

PROP. 3.9 *Let V be the valuation function of Henkin model H. The following properties hold for all variable assignments ϕ, terms $s_o, t_o, l_\alpha, r_\alpha$, and variables X_α, W_α: $V(\phi, \neg s_o) = T$ iff $V(\phi, s_o) = F$, $V(\phi, s_o \vee t_o) = T$ iff $V(\phi, s_o) = T$ or $V(\phi, t_o) = T$, $V(\phi, s_o \rightarrow t_o) = T$ iff $V(\phi, s_o) = F$ or $V(\phi, t_o) = T$, $V(\phi, \forall X_\alpha s_o) = V(\phi, \Pi_{(\alpha\rightarrow o)\rightarrow o}\lambda X_\alpha s_o) = T$ iff for all $v \in D_\alpha$ holds $V(\phi[w/W_\alpha], (\lambda X_\alpha s_o)\, W_\alpha) = T$, and if $l_\alpha =_{\beta\eta} r_\alpha$ then $V(\phi, l_\alpha) = V(\phi, r_\alpha)$.*

4 Quantified Conditional Logics in HOL

QCL formulas are identified with certain HOL terms (predicates) of type $(i \rightarrow o)$. They can be applied to terms of type i, which are assumed to denote possible worlds. Type $(i \rightarrow o)$ is abbreviated as τ in the remainder.

DEF. 4.1 (Embedding of QCL in HOL) *The mapping $\lfloor\cdot\rfloor$ identifies QCL formulas φ with HOL terms $\lfloor\varphi\rfloor$ of type τ. The mapping is recursively defined:*

$$\begin{array}{lcl}
\lfloor P \rfloor & = & P_\tau \\
\lfloor k(X^1, \ldots, X^n) \rfloor & = & k_{u^n\rightarrow\tau}\, X^1_u \ldots X^n_u \\
\lfloor \neg\varphi \rfloor & = & \neg_{\tau\rightarrow\tau}\, \lfloor\varphi\rfloor \\
\lfloor \varphi \vee \psi \rfloor & = & \vee_{\tau\rightarrow\tau\rightarrow\tau}\, \lfloor\varphi\rfloor \lfloor\psi\rfloor \\
\lfloor \varphi \Rightarrow \psi \rfloor & = & \Rightarrow_{\tau\rightarrow\tau\rightarrow\tau}\ \lfloor\varphi\rfloor \lfloor\psi\rfloor \\
\lfloor \forall^{co} X \varphi \rfloor & = & \Pi^{co}_{(u\rightarrow\tau)\rightarrow\tau}\, \lambda X_u \lfloor\varphi\rfloor \\
\lfloor \forall^{va} X \varphi \rfloor & = & \Pi^{va}_{(u\rightarrow\tau)\rightarrow\tau}\, \lambda X_u \lfloor\varphi\rfloor \\
\lfloor \forall P \varphi \rfloor & = & \Pi_{(\tau\rightarrow\tau)\rightarrow\tau}\, \lambda P_\tau \lfloor\varphi\rfloor
\end{array}$$

P_τ and $X^1_u, \ldots, X^n_u$ are variables and $k_{u^n\rightarrow\tau}$ is a constant symbol. $\neg_{\tau\rightarrow\tau}$, $\vee_{\tau\rightarrow\tau\rightarrow\tau}$, $\Rightarrow_{\tau\rightarrow\tau\rightarrow\tau}$, $\Pi^{co,va}_{(u\rightarrow\tau)\rightarrow\tau}$ and $\Pi_{(\tau\rightarrow\tau)\rightarrow\tau}$ realize the QCL connectives in HOL. They abbreviate the following HOL terms:[1]

$$\begin{array}{ll}
\neg_{\tau\rightarrow\tau} & = \lambda A_\tau \lambda X_i \neg(A\, X) \\
\vee_{\tau\rightarrow\tau\rightarrow\tau} & = \lambda A_\tau \lambda B_\tau \lambda X_i (A\, X \vee B\, X) \\
\Rightarrow_{\tau\rightarrow\tau\rightarrow\tau} & = \lambda A_\tau \lambda B_\tau \lambda X_i \forall V_i (f\, X\, A\, V \rightarrow B\, V) \\
\Pi^{co}_{(u\rightarrow\tau)\rightarrow\tau} & = \lambda Q_{u\rightarrow\tau} \lambda V_i \forall X_u (Q\, X\, V) \\
\Pi^{va}_{(u\rightarrow\tau)\rightarrow\tau} & = \lambda Q_{u\rightarrow\tau} \lambda V_i \forall X_u (eiw\, V\, X \rightarrow Q\, X\, V) \\
\Pi_{(\tau\rightarrow\tau)\rightarrow\tau} & = \lambda R_{\tau\rightarrow\tau} \lambda V_i \forall P_\tau (R\, P\, V)
\end{array}$$

Constant symbol f in the mapping of $\Rightarrow$ is of type $i \rightarrow \tau \rightarrow \tau$. It realizes the selection function. Constant symbol eiw (for 'exists in world'), which is of type $i \rightarrow u \rightarrow o$, is associated with the varying domains. The interpretations of f and eiw are chosen appropriately below, cf. Def. 5.2. Moreover, for the varying domains a non-emptiness axiom is postulated:

$$\forall W_i \exists X_u (eiw\, W\, X) \qquad \text{(NE)}$$

The above mapping induces mappings $\lfloor IV \rfloor$, $\lfloor PV \rfloor$ and $\lfloor SYM \rfloor$ of the sets IV, PV and SYM respectively.

DEF. 4.2 (Embedding of the notion of validity) *Analyzing the validity of a translated formula $\lfloor\varphi\rfloor$ for a world represented by term t_i corresponds to evaluating the application $(\lfloor\varphi\rfloor\, t_i)$. In line with Benzmüller et al. [2012a], validity is thus defined as $vld_{\tau\rightarrow o} = \lambda A_\tau \forall S_i (A\, S)$. With this definition, validity of a QCL formula φ in QCK corresponds to the validity of $(vld\, \lfloor\varphi\rfloor)$ in HOL, and vice versa. Further semantic notions such as countersatisfiable $:= \lambda A_\tau.\neg\forall S_i.(A\, S)$ can be introduced.*

Note that the definition of $\Rightarrow$ can be further generalized as follows:

$$\Rightarrow\, = \lambda f_{i\rightarrow\tau\rightarrow\tau} \lambda A_\tau \lambda B_\tau \lambda X_i \forall V_i (f\, X\, A\, V \rightarrow B\, V)$$

Now, several selection function symbols $f^1, \ldots, f^n$ can be introduced in the signature and $\Rightarrow$ can be instantiated with each of them leading to different, indexed conditional operators $\Rightarrow_{f^1}, \ldots, \Rightarrow_{f^n}$. These indexed conditionals can now easily be associated with different axioms and properties, and they can be combined.

[1] Note the predicate argument A of f in the term for $\Rightarrow_{\tau\rightarrow\tau\rightarrow\tau}$ and the second-order quantifier $\forall P_\tau$ in the term for $\Pi_{(\tau\rightarrow\tau)\rightarrow\tau}$. FOL encodings of both constructs, if feasible, will be less natural.

5 Soundness and Completeness

To prove the soundness and completeness of the above embedding, a mapping from QCL models into Henkin models is employed. This mapping utilizes a corresponding mapping of QCL variable assignments into HOL variable assignments.

DEF. 5.1 (Mapping of Assignments) *Let* $g = (g^i : IV \longrightarrow D,\ g^p : PV \longrightarrow Q)$ *be a variable assignment for QCL. The corresponding variable assignment* $\lfloor g \rfloor = (\lfloor g^i \rfloor : \lfloor IV \rfloor \longrightarrow D,\ \lfloor g^p \rfloor : \lfloor PV \rfloor \longrightarrow Q)$ *for HOL is defined such that* $\lfloor g \rfloor(X_u) = \lfloor g \rfloor(\lfloor X \rfloor) = g(X)$ *and* $\lfloor g \rfloor(P_\tau) = \lfloor g \rfloor(\lfloor P \rfloor) = g(P)$ *for all* $X_u \in \lfloor IV \rfloor$ *and* $P_\tau \in \lfloor PV \rfloor$*. Finally,* $\lfloor g \rfloor$ *is extended to an assignment for variables* Z_α *of arbitrary type by choosing* $\lfloor g \rfloor(Z_\alpha) = d \in D_\alpha$ *arbitrary, if* $\alpha \neq u, \tau$.

DEF. 5.2 (Henkin model H^M) *Given a QCL model* $M = \langle S, f, D, D', Q, I \rangle$*. The Henkin model* $H^M = \langle \{D_\alpha\}_{\alpha \in T}, I \rangle$ *for* M *is defined as follows:* D_i *is chosen as the set of possible worlds* S*,* D_u *is chosen as the first-order domain* D *(cf. definition of* $\lfloor g^i \rfloor$ *and the remark in Def. 2.5),* D_τ *is chosen as the set of sets of possible worlds* Q *(cf. definition of* $\lfloor g^p \rfloor$*)*[2]*, and all other sets* $D_{\alpha \to \beta}$ *are chosen as (not necessarily full) sets of functions from* D_α *to* D_β*. For all* $D_{\alpha \to \beta}$ *the rule that every term must have a denotation must be obeyed, in particular, it is required that* $D_{u^n \to \tau}$*,* $D_{i \to \tau \to \tau}$ *and* $D_{i \to u \to o}$ *contain the elements* $Ik_{u^n \to \tau}$*,* $If_{i \to \tau \to \tau}$ *and* $Ieiw_{i \to u \to o}$ *as characterized next. Interpretation* I *is constructed as follows:*

1. *Let* $k_{u^n \to \tau} = \lfloor k \rfloor$ *for* n*-ary* $k \in SYM$ *and let* $X^i_u = \lfloor X^i \rfloor$ *for* $X^i \in IV$*,* $i = 1, \ldots, n$*.* $Ik_{u^n \to \tau} \in D_{u^n \to \tau}$ *is chosen such that* $(I\, k_{u^n \to \tau})(\lfloor g \rfloor(X^1_u), \ldots, \lfloor g \rfloor(X^n_u), w) = T$ *for all worlds* $w \in D_i$ *with* $M, g, w \models k(X^1, \ldots, X^n)$*, i.e., if* $\langle g^i(X^1), \ldots, g^i(X^n) \rangle \in I(k, w)$*. Otherwise,* $(I\, k_{u^n \to \tau})(\lfloor g \rfloor(X^1_u), \ldots, \lfloor g \rfloor(X^n_u), w) = F$.
2. $I\, f_{i \to \tau \to \tau} \in D_{i \to \tau \to \tau}$ *is chosen such that* $(I f_{i \to \tau \to \tau})(s, q, t) = T$ *for all worlds* $s, t \in D_i$ *and* $q \in D_\tau$ *with* $t \in f(s, \{x \in S \mid q(x) = T\})$ *in* M*. Otherwise,* $(I f_{i \to \tau \to \tau})(s, q, t) = F$.
3. $I\, eiw_{i \to u \to o} \in D_{i \to u \to o}$ *is chosen such that* $(I\, eiw_{i \to u \to o})(s, d) = T$ *for individuals* $d \in D'(s)$ *in* M*. Otherwise,* $(I\, eiw_{i \to u \to o})(s, d) = F$.
4. *For all other constants* c_α*, choose* Ic_α *arbitrary.*[3]

It is not hard to verify that H^M *is a Henkin model.*[4]

LEMMA 5.3 *Let* H^M *be a Henkin model for a QCL model* M *which validates axiom NE. For all quantified conditional logic formulas* δ*, variable assignments* g *and worlds* s *it holds:* $M, g, s \models \delta$ *iff* $V(\lfloor g \rfloor[s/S_i], \lfloor \delta \rfloor\, S_i) = T$.

Proof: *The proof is by induction on the structure of* δ.

The cases for $\delta = P$*,* $\delta = k(X^1, \ldots, X^n)$*,* $\delta = \neg\varphi$*,* $\delta = \varphi \vee \psi$*, and* $\delta = \varphi \Rightarrow \psi$ *are similar to Benzmüller et al. [2012a], Lemma 1.*

If $\delta = \forall^{va} X \varphi$*, by definition, it holds* $M, g, s \models \forall^{va} X \varphi$ *iff for all* $d \in D'(s)$ *we have* $M, ([d/X_u]g^i, g^p), s \models \varphi$*. By induction the latter condition is equivalent to: for all* $d \in D'(s)$ *holds* $V(\lfloor([d/X_u]g^i, g^p)\rfloor[s/S_i], (\lfloor \varphi \rfloor\, S_i)) = V(\lfloor g \rfloor[s/S_i][d/X_u], \lfloor \varphi \rfloor\, S_i) = T$*. Due to the choice of* $I\, eiw$ *in* H^M*, the non-emptiness condition NE and Prop. 3.9 this is equivalent to: for all* $d \in D$ *it holds* $V(\lfloor g \rfloor[s/S_i][d/X_u], eiw\, S_i\, X_u \to \lfloor \varphi \rfloor\, S_i)$*. Hence, by Prop. 3.9 we have that* $V(\lfloor g \rfloor[s/S_i], \Pi_{(u \to o) \to o} \lambda X_u (eiw\, S_i\, X_u \to \lfloor \varphi \rfloor\, S_i)) =_{\beta\eta} V(\lfloor g \rfloor[s/S_i], (\lambda W_i\, \Pi_{(u \to o) \to o}\, \lambda X_u (eiw\, W_i\, X_u \to \lfloor \varphi \rfloor\, W_i))\, S_i) = T$*. By def. of* $\Pi^{va}_{(u \to \tau) \to \tau}$*, def. of* $\lfloor . \rfloor$ *and Prop. 3.9 we get* $V(\lfloor g \rfloor[s/S_i], (\Pi^{va}_{(u \to \tau) \to \tau}\ \lambda X_u \lfloor \varphi \rfloor)\, S_i) = V(\lfloor g \rfloor[s/S_i], \lfloor \forall^{va} X \varphi \rfloor\, S_i) = T$.

The cases for $\delta = \forall^{co} X \varphi$ *and* $\delta = \forall P \varphi$ *are similar (actually simpler).* □

THEOREM 5.4 (Soundness and Completeness) $\{NE\} \models vld\, \lfloor \varphi \rfloor$ *in HOL iff* $\models \varphi$ *in QCK*

Proof: *(Soundness,* $\rightarrow$*) The proof is by contraposition. Assume* $\not\models \varphi$ *in QCK, i.e, there is a QCL model* $M = \langle S, f, D, D', Q, I \rangle$*, an assignment* g *and a world* $s \in S$*, such that* $M, g, s \not\models \varphi$*. By Lemma 5.3 it holds that* $V(\lfloor g \rfloor[s/S_i], \lfloor \varphi \rfloor\, S_i) = F$ *in Henkin model* $H^M = \langle \{D_\alpha\}_{\alpha \in T}, I \rangle$ *for* M*. Furthermore,* $H^M \models \{NE\}$*. Thus, by Prop. 3.9, definition of vld and since* $\forall S_i(\lfloor \varphi \rfloor\, S_i) =_{\beta\eta} vld\, \lfloor \varphi \rfloor$ *it holds that* $V(\lfloor g \rfloor, \forall S_i(\lfloor \varphi \rfloor\, S_i)) = V(\lfloor g \rfloor, vld\, \lfloor \varphi \rfloor) = F$*. Hence,* $H^M \not\models vld\, \lfloor \varphi \rfloor$*, and thus* $\{NE\} \not\models vld\, \lfloor \varphi \rfloor$ *in HOL.*

(Completeness, $\leftarrow$*) The proof is again by contraposition. Assume* $\{NE\} \not\models vld\, \lfloor \varphi \rfloor$ *in HOL, i.e., there is a Henkin model* $H = \langle \{D_\alpha\}_{\alpha \in T}, I \rangle$ *and an assignment* ϕ *such that* $H \models \{NE\}$ *and* $V(\phi, vld\, \lfloor \varphi \rfloor) = F$*. Without loss of generality it can be assumed that Henkin model* H *is in fact a Henkin model* H^M *for a corresponding QCL model* M *and that* $\Phi = \lfloor g \rfloor$ *for a corresponding QCL variable assignment* g*. By Prop. 3.9 and since* $vld\, \lfloor \varphi \rfloor =_{\beta\eta} \forall S_i(\lfloor \varphi \rfloor\, S_i)$ *it holds* $V(\lfloor g \rfloor, \forall S_i(\lfloor \varphi \rfloor\, S_i)) = F$*, and hence, by def. of vld,* $V(\lfloor g \rfloor[s/S_i], \lfloor \varphi \rfloor\, S_i) = F$ *for some* $s \in D$*. By Lemma 5.3* $M, g, s \not\models \varphi$*, and hence* $\not\models \varphi$ *in QCK.* □

This shows that QCL is a natural fragment of HOL.

6 Object- and meta-level automation of QCL

Figure 1 presents the HOL encoding of QCL connectives and axiom NE in THF syntax; THF is a concrete syntax for HOL [Sutcliffe and Benzmüller, 2010].[5]

[2] Sets are identified with their characteristic functions.

[3] In fact, it may be safely assumed that there are no other typed constant symbols given, except for the symbols $f_{i \to \tau \to \tau}$, $eiw_{i \to u \to o}$, $k_{u^n \to \tau}$, and the logical connectives.

[4] In H^M we have merely fixed $D_i = S$, $D_u = D$ and $D_\tau = Q$, and the interpretations of $k_{u^n \to \tau}$, $f_{i \to \tau \to \tau}$ and $eiw_{i \to u \to o}$. These choices are not in conflict with any of the requirements regarding frames and interpretations. The existence of a valuation function V for an HOL interpretation crucially depends on how sparse the function spaces have been chosen in frame $\{D_\alpha\}_{\alpha \in T}$. [Andrews, 1972a] discusses criteria that are sufficient to ensure the existence of a valuation function; they require that certain λ-abstractions have denotations in frame $\{D_\alpha\}_{\alpha \in T}$. Since it is explicitly required that every term denotes and since Q has been appropriately constrained in QCL models (and hence $D_{i \to o}$ in H^M) these requirements are met.

[5] Notes on THF: `$i` and `$o` represent the HOL base types i and o. `$i>$o` encodes a function (predicate) type. Function or predi-

```
%---- file: Axioms.thf -----------------------------------
%--- type mu for individuals
thf(mu,type,(mu:$tType)).
%--- reserved constant for selection function f
thf(f,type,(f:$i>($i>$o)>$i>$o)).
%--- ‘exists in world’ predicate for varying domains;
%--- for each v we get a non-empty subdomain eiw@v
thf(eiw,type,(eiw:$i>mu>$o)).
thf(nonempty,axiom,(![V:$i]:?[X:mu]:(eiw@V@X))).
%--- negation, disjunction, material implication
thf(not,type,(not:($i>$o)>$i>$o)).
thf(or,type,(or:($i>$o)>($i>$o)>$i>$o)).
thf(impl,type,(impl:($i>$o)>($i>$o)>$i>$o)).
thf(not_def,definition,(not = (^[A:$i>$o,X:$i]:~(A@X)))).
thf(or_def,definition,(or
 = (^[A:$i>$o,B:$i>$o,X:$i]:((A@X)|(B@X))))).
thf(impl_def,definition,(impl
 = (^[A:$i>$o,B:$i>$o,X:$i]:((A@X)=>(B@X))))).
%--- conditionality
thf(cond,type,(cond:($i>$o)>($i>$o)>$i>$o)).
thf(cond_def,definition,(cond
 = (^[A:$i>$o,B:$i>$o,X:$i]:![W:$i]:((f@X@A@W)=>(B@W))))).
%--- quantification (constant dom., varying dom., prop.)
thf(all_co,type,(all_co: (mu>$i>$o)>$i>$o)).
thf(all_va,type,(all_va:(mu>$i>$o)>$i>$o)).
thf(all,type,(all:(($i>$o)>$i>$o)>$i>$o)).
thf(all_co_def,definition,(all_co
 = (^[A:mu>$i>$o,W:$i]:![X:mu]:(A@X@W)))).
thf(all_va_def,definition,(all_va
 = (^[A:mu>$i>$o,W:$i]:![X:mu]:((eiw@W@X)=>(A@X@W))))).
thf(all_def,definition,(all
 = (^[A:($i>$o)>$i>$o,W:$i]:![P:$i>$o]:(A@P@W)))).
%--- box operator based on conditionality
thf(box,type,(box:($i>$o)>$i>$o)).
thf(box_def,definition,(box
 = (^[A:$i>$o]:(cond@(not@A)@A)))).
%--- notion of validity of a conditional logic formula
thf(vld,type,(vld:($i>$o)>$o)).
thf(vld_def,definition,(vld
 = (^[A:$i>$o]:![S:$i]:(A@S)))).
%---- end file: Axioms.thf -------------------------------
```

Figure 1: THF encoding of QCL.

The content of Fig. 1 is stored in file Axioms.ax and this file is imported in all example problems discussed below. Further definitions, e.g., for the connectives $\wedge$ or $\exists^{va}$, can be easily added.

Following Sec. 4 and by employing the QCL connectives defined in Axioms.ax, QCL statements can be directly expressed and automated in HOL. As will be illustrated below, this includes both statements formulated at the QCL object-level and statements at the QCL meta-level, i.e., statements about properties of QCL. In both cases the reasoning is carried out by off-the-shelf automated theorem provers and model finders for HOL. In the experiments discussed below the following systems have been employed: SATALLAX (version 2.6) [Brown, 2012], ISABELLE (version 2012) [Nipkow *et al.*, 2002], LEO-II [Benzmüller *et al.*, 2008] (version 1.5.0), NITPICK (version 2012) [Blanchette and Nipkow, 2010] and AGSYHOL (version 1.0) [Lindblad, 2012].

```
%--------------------------------------------------------------
include('Axioms.ax').
%--- we (falsely) conjecture: ID is implied by Axioms.ax
thf(id,conjecture,(vld@(all@^[P:$i>$o]:(cond@P@P)))).
%--------------------------------------------------------------
```

Figure 2: THF encoding of ‘the validity of ID is implied’.

In the remainder they will be abbreviated with S, I, L, N and A, respectively. These systems work for Henkin semantics and they support the THF syntax as a common input language. Moreover, the SystemOnTPTP infrastructure [Sutcliffe, 2009] enables remote calls to instances of these provers at the University of Miami. Exploiting these features a simple shell script has been written that bundles these systems into a HOL meta-prover, called H in the remainder. H has been employed in the experiments reported below. Time measurements are given for the subsystems of H, e.g., (H_N=5.2, H_S=0.0) expresses that Nitpick and Satallax have terminated (with a result as mentioned) within 5.2 seconds, respectively 0.0 seconds, of CPU time.[6] If respective information is not given for a subprover then it produced 'Unknown' or 'Timeout' as a result. In the experiments each subprover of H was given a 20s timeout.

Automation of meta-level reasoning. The HOL approach supports meta-level reasoning for QCL. This includes proving the independence or dependence of QCL axioms, exploring correspondences between axioms and associated properties of the selection function, and showing the admissibility of inference rules. Respective examples for propositional CLs have been carried out by Benzmüller et al. [2012a]. Below the focus is on meta-level properties of QCL.

EXAMPLE 1 (Consistency) *H quickly proves the satisfiability of* Axioms.thf *($H_{L,S}$=0.0, H_N=6.4). This confirms the consistency of QCL relative to the consistency of metalogic HOL.*

EXAMPLE 2 (Axioms and rules) *Prominent CL axioms are the law of identity ID: $\forall P\,(P \Rightarrow P)$ and modus ponens for $\Rightarrow$ MP: $\forall P \forall Q\,(P \Rightarrow Q) \rightarrow (P \rightarrow Q)$. Further prominent axioms include CS, CEM, AC, RT, CV, CA; cf. Pozzato [2010]). H proves the independence of these axioms by generating countermodels to respective dependence conjectures as exemplified for ID in Fig. 2, which encodes that $vld_{\tau\rightarrow o} \forall P_\tau (P_\tau \Rightarrow_{\tau\rightarrow\tau\rightarrow\tau} P_\tau)$ is implied by the theory in* Axioms.thf*. When receiving this input file, the subprovers of H expand the definitions contained in* Axioms.thf *and apply their internal reasoning procedures to search for a solution. Countermodels are generated quickly (ID/MP: $H_{L,S}$=0.0, H_N=6.4). Similarly, the admissibility of the rules RCEA, RCK and RCEC for QCL can be quickly proved by H. That is, these rules are automatically entailed in the HOL approach and can therefore be omitted without loosing completeness (cf. [Benzmüller* et al.*, 2012a], Problem 2).*

EXAMPLE 3 (Properties of QCL) *Regarding quantifiers properties as given next are of some interest (where*

cate application as in $(eiw\,V\,X)$ is encoded as ((eiw@V)@X) or simply as (eiw@V@X), i.e., function application is represented by @. Universal quantification and λ-abstraction as in $\lambda A_{i\rightarrow o} \forall S_i (A\,S)$ and are represented as in ^[X:$i>$o]:![S:$i]:(A@S); ? is the existential quantifier, and $\neg, \vee, \wedge, and \Rightarrow$ (mat. impl.) are written as , |, &, and =>. Comments begin with %. Better formatted and easier readable presentations of the THF code in this paper can easily be generated with the TPTP tools of Sutcliffe [2009]; here the minimization of space has been of primary interest.

[6] Specifications of the SystemOnTPTP computers at Miami: i686 Intel(R) Xeon(R) CPU 5140 @ 2.33GHz, NumberOfCPUs: 4, RAMPerCPU: 1006.25MB, OS: Linux 2.6.32.26.

```
%---------------------------------------------------------
include('Axioms.ax').
%--- axioms ID and MP
thf(id,axiom,(vld@(all@^[P:$i>$o]:(cond@P@P)))).
thf(mp,axiom,
 (vld@(all@^[P:$i>$o]:(all@^[Q:$i>$o]:
  (impl@(cond@P@Q)@(impl@P@Q)))))).
%--- type declarations
thf(horse,type,(horse:mu>$i>$o)).
thf(wings,type,(wings:mu>$i>$o)).
thf(fly,type,(fly:mu>$i>$o)).
thf(pegasus,type,(pegasus:mu)).
%--- the statements
thf(ax1,axiom,
 (vld@(all_va@^[X:mu]:(impl@(horse@X)@(not@(wings@X)))))).
thf(ax2,axiom,(vld@(horse@pegasus))).
thf(ax3,axiom,(vld@(wings@pegasus))).
%---------------------------------------------------------
```

Figure 3: THF encoding of Example 4.

$* \in \{co, va\}$)*; P1 and P2 are known as Barcan and converse Barcan formula.*

$$\forall^* X(\varphi \Rightarrow \psi(X)) \rightarrow (\varphi \Rightarrow \forall^* X\psi(X)) \quad \text{(P1)}$$

$$(\varphi \Rightarrow \forall^* X\psi(X)) \rightarrow \forall^* X(\varphi \Rightarrow \psi(X)) \quad \text{(P2)}$$

$$\forall^* X(\varphi(X) \rightarrow \psi(X)) \rightarrow (\forall^* X\varphi(X) \rightarrow \forall^* X\psi(X)) \quad \text{(P3)}$$

In P1 and P2 it is assumed that X does not occur free in φ. Instances of P1 and P2 can be formulated in HOL. E.g., for P1-$\forall^{va}$ the resulting HOL formula is vld $(\forall^{va}X(a \Rightarrow (b\,X)) \rightarrow (a \Rightarrow \forall^{va}X(b\,X)))$, where a and b are constants of types τ and $u \rightarrow \tau$, respectively. Such instances of P1-$\forall^{co}$ and P2-$\forall^{co}$ can be quickly proved by H (P1-$\forall^{co}$/P2-$\forall^{co}$: $H_{A,L,S}$=0.0,H_I=5.8). This is consistent with the assumption of constant domain semantics for $\forall^{co}$. As expected, H reports countersatisfiability for P1-$\forall^{va}$ and P2-$\forall^{va}$ (P1-$\forall^{va}$/P2-$\forall^{va}$: H_S=0.0,H_L=0.4,H_N=6.4). This is illustrated for P1-$\forall^{va}$, which expands into HOL formula P1' :=

$$\forall S_i(\neg\forall X_u(\neg\, eiw\; S\; X \vee \forall V_i(\neg f\; S\; a\; V \vee b\; X\; V))\vee$$
$$\forall W_i(\neg f\; S\; a\; W \vee \forall Y_u(\neg\, eiw\; W\; Y \vee b\; Y\; W)))$$

The countermodel generated by Nitpick is: Let $D_u = \{u^1, u^2\}$, $D_i = \{i^1, i^2\}$, $I(a) = \emptyset$, $I(b) = \{(u^2, i^2)\}$, $I(eiw) = \{(i^1, u^2), (i^2, u^1)\}$, and $I(f) = \{(i^1, \emptyset, i^2)\}$. Now, choose i^1 for S, i^2 for W, and u^1 for Y. This invalidates P1', and hence P1. Note that the varying domains $D'(i^1) = \{u^2\}$ and $D'(i^2) = \{u^1\}$ (cf. $I(eiw)$) play a crucial role.

P3 is a standard property of first-order logic. H proves its validity for $\forall^{va}$ and $\forall^{co}$ (P3-$\forall^{va}$/$\forall^{co}$: $H_{A,L,S}$=0.0,H_I=5.9).

Automation of object-level reasoning. Automation of object-level reasoning is illustrated by prominent examples from the literature; cf. Delgrande [1998], Sec. 5.1. Following Delgrande, let $\phi \Rightarrow_X \psi$ be an abbreviation for $(\exists^{va}X\phi) \Rightarrow \forall^{va}X(\phi \rightarrow \psi)$. Moreover, in all examples below axioms ID and MP are postulated, i.e, the reference logic is QCK+ID+MP.

EXAMPLE 4 (Pegasus, the winged horse) *It can be consistently stated that horses (h) contingently do not have wings (w) but Pegasus (p) is a winged horse.*

$$\forall^{va}X(h(X) \rightarrow \neg w(X)), \quad h(p), \quad w(p) \qquad \text{(E1)}$$

A THF encoding of E1 is presented in Fig. 3. H confirms the satisfiability of E1 (with H_N=7.7). The finite model generated by Nitpick tells us that Pegasus is not 'actual', i.e., does not exist (cf. eiw) in any world. As expected, when E1 is formulated with $\forall^{co}$ instead of $\forall^{va}$ then H reports unsatisfiability ($H_{L,S}$=0.0, H_I=5.8).

EXAMPLE 5 (Opus, the penguin) *Birds (b) normally fly (f), but Opus (o) is a bird that normally does not fly.*

$$b(X) \Rightarrow_X f(X), \quad b(o), \quad b(o) \Rightarrow \neg f(o) \qquad \text{(E2)}$$

This can be consistently stated in QCK+ID+MP and H reports a finite model (H_N=8.6). When $\forall^{co}$ is used instead of $\forall^{va}$, H reports unsatisfiability for E2 (H_S=0.0, H_I=7.9).

It can also be consistently asserted that birds normally fly and that necessarily Opus the bird does not fly.

$$b(X) \Rightarrow_X f(X), \quad \Box(b(o) \wedge \neg f(o)) \qquad \text{(E3)}$$

Again, H reports a finite model (H_N=8.7). H reports unsatisfiability when $\forall^{co}$ is used in E3 (H_S=0.0, H_I=7.6).

However, it cannot be consistently asserted that birds normally fly and that necessarily there is a non-flying bird.

$$b(X) \Rightarrow_X f(X), \quad \Box\exists^{va}(b(X) \wedge \neg f(X)) \qquad \text{(E4)}$$

H reports unsatisfiability (H_S=0.0, H_I=8.7). H also reports unsatisfiability when $\forall^{co}$ is used in E4 (H_S=0.0, H_I=8.8).

Finally—as desired—it can be consistently stated that birds normally fly, penguins normally do not fly and that all penguins are necessarily birds.

$$b(X) \Rightarrow_X f(X), \quad p(X) \Rightarrow_X \neg f(X),$$
$$\forall^{va}\Box(p(X) \rightarrow b(X)) \qquad \text{(E5)}$$

H generates a finite model (E4-$\forall^{va}$: H_N=8.8; E4-$\forall^{co}$: H_N=7.9). Moreover, H can conclude from the statements in E5 that birds are normally not penguins (E6-$\forall^{va}$: H_S=0.9, H_L=10.2, H_A=9.4; E6-$\forall^{co}$: H_S=0.8, H_L=10.1, H_A=0.3).

$$E5 \vdash b(X) \Rightarrow_X \neg p(X) \qquad \text{(E6)}$$

On the other hand, and in line with Delgrande, H reports a countermodel for the following statement when $\forall^{va}$ is used (H_N=8.7).

$$E5 \vdash b(o) \Rightarrow \neg p(o) \qquad \text{(E7)}$$

When $\forall^{co}$ is used, H reports a theorem (H_S=0.8, H_A=0.4).

A closer look at the performance of H in this section seems to indicate that Satallax and Nitpick subsume the other subprovers of H. However, experiments in related areas show that there are many counterexamples to this conjecture (cf. the recent case study in first-order modal logics as reported by [Benzmüller and Raths, 2013]). The reason is that the search space in HOL is usually highly branching and that the current HOL systems often employ quite orthogonal search strategies and heuristics. In fact, Isabelle, the reigning CASC[7] champion in higher-order automated theorem proving has not been the strongest prover here (e.g. it failed on E6). It is thus recommendable to use the existing HOL provers in combination, just as illustrated in this paper.

[7] The results of the 2012 CASC competition are available at `http://www.cs.miami.edu/~tptp/CASC/J6/`.

7 Related work

Delgrande [1998] motivates the development of first-order CLs and he points to the problems of naive (constant domain) quantification in this context (a selection of his examples have been replicated with prover H above). The framework developed by Delgrande supports variable and constant domain quantification; it does not address propositional quantification. Regarding the semantics of $\Rightarrow$, Delgrande [1998] utilizes ternary accessibility relation. Earlier he also worked with selection function semantics [Delgrande, 1987].

First-order quantification in CLs has been studied also by Friedman et al. [2000]. Their focus is on default reasoning and they develop a subjective and statistical first-order logic of conditionals. Their conditional logic semantics is based on plausibility measures. Unfortunately, an implementation of their framework is not available. This also holds for Delgrande's approach.

Further alternative approaches to first-order nonmonotonic reasoning have been presented by Lehman & Magidor [1990] and Schlechta [1995].

The HOL approach supports the automation of arbitrary extensions of normal conditional logics. This is possible by simply postulating respective axioms (cf. ID and MP in Sec. 6). For this, quantification over propositional variables is crucial, which is not supported in any of the above approaches. Alternatively, corresponding semantic conditions on the selection function can be postulated in the HOL approach (if they exist and are known). Moreover, due to the embedding of QCL in the HOL meta-logic, meta-logical reasoning about QCL properties is facilitated. This is not the case for the above approaches.

The HOL approach supports and exploits explicit term representations of possible worlds/states. Hence, it can easily be extended to also cover hybrid CLs and their first-order extensions [Myers *et al.*, 2009].

Even for propositional CLs very few provers currently exist. Some sequent and tableaux calculi are available, e.g., for logics CK+{ID, MP/CEM, CS} and CK+{CEM, MP} [Pozzato, 2010; Schröder *et al.*, 2010]. However, no theorem provers are, e.g., known for CK+{CS, AC, RT, CV, CA} and no model builders for CK+{ID, CS, AC, RT, CV, CA}. The work presented here also supports those (and other) combinations and their extension to QCLs.

An embedding of propositional CLs in HOL has been presented by Benzmüller et al. [2012a]. First-order constant domain quantification has been studied by Benzmüller and Genovese [2011]. However, as Delgrande's examples show, constant domain quantification is counterintuitive in conditional logics. First-order and propositional quantification has also been addressed in a related embedding of quantified multimodal logic in HOL [Benzmüller and Paulson, 2013]; again, only constant domain quantification has been considered there. Varying and cumulative domain quantification have subsequently been added and Benzmüller et al. [2012b] and Benzmüller and Raths [2013] have carried out comparative evaluations of respective provers for quantified monomodal logic with very encouraging results for the HOL approach.

8 Conclusion

First, a notion of quantified conditional logic has been provided that includes first-order and propositional quantification, the former with respect to varying and constant domain semantics. Second, a sound and complete embedding of this logic in classical higher-order logic (w.r.t. Henkin semantics) has been presented. Third, this logic has been automated with off-the-shelf higher-order automated theorem provers and it has been shown that prominent challenges in nonmonotonic reasoning can be adequately addressed this way. The offered automation includes theorem proving and model finding and it is applicable to object-level and meta-level reasoning.

Note the particular relevance of the second point: Since quantified conditional logics are natural fragments of classical higher-order logic important meta-theoretical resp. proof-theoretical results can be obtained for free, including cut-elimination and compactness. This is because these results are already known for classical higher-order logic (see e.g. the cut-free sequent calculus of Benzmüller et al. [2009]). Consequently, these results also apply to the embedded logics. In fact, my current program to study a wide range of classical and non-classical logics and their combinations as fragments of HOL can be seen as an elegant model theoretic approach to achieve such results.

A comparative evaluation of the presented work has been impossible so far, since there are no other provers for quantified conditional logics practically available to date. However, the results of the experiments so far are promising. Moreover, higher-order automated theorem provers have shown significant improvements over the last years and further improvements are to be expected. The conjecture therefore is that the approach will eventually scale also for more complex examples. An investigation of this conjecture is future work.

Future work will also study combinations of quantified conditional logic with other classical and nonclassical logics. Such combinations are relevant, e.g., for the adequate treatment of nonmonotonic concepts in expressive ontologies such as SUMO [Niles and Pease, 2001; Pease, 2011] or Cyc [Matuszek *et al.*, 2006].

Acknowledgments: I thank the implementors of the HOL reasoning systems employed in this work, in particular, Jasmin Blanchette and Chad Brown. Moreover, I thank Geoff Sutcliffe for providing and maintaining the TPTP infrastructure. His support has been one crucial factor for the good progress of HOL ATP in recent years. I also thank Valerio Genovese, Dov Gabbay and Daniele Rispoli. They have pointed me to conditional logics and we have fruitfully collaborated on the propositional level. Valerio Genovese has also provided some support w.r.t. the extension of this work to constant domain first-order quantification.

References

[Andrews, 1972a] P.B. Andrews. General Models, Descriptions, and Choice in Type Theory. *Journal of Symbol Logic*, 37(2):385–394, 1972.

[Andrews, 1972b] P.B. Andrews. General models and extensionality. *Journal of Symbol Logic*, 37(2):395–397, 1972.

[Andrews, 2009] P.B. Andrews. Church's type theory. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Stanford University, spring 2009 edition, 2009.

[Benzmüller and Genovese, 2011] C. Benzmüller and V. Genovese. Quantified conditional logics are fragments of HOL. In *The Intl. Conf. on Non-classical Modal and Predicate Logics (NCMPL)*, Guangzhou (Canton), China, 2011. The paper is available as arXiv:1204.5920v1.

[Benzmüller and Paulson, 2013] C. Benzmüller and L.C. Paulson. Quantified multimodal logics in simple type theory. *Logica Universalis*, 7(1):7-20, 2013.

[Benzmüller and Raths, 2013] C. Benzmüller and T. Raths. HOL based automation of first-order modal logics. *Intl. Workshop on Methods for Modalities*, 2013, submitted.

[Benzmüller *et al.*, 2008] C. Benzmüller, F. Theiss, L. Paulson, and A. Fietzke. LEO-II - a cooperative automatic theorem prover for higher-order logic. In *Proc. of IJCAR 2008*, vol. 5195 of *LNCS*, pp. 162–170. Springer, 2008.

[Benzmüller *et al.*, 2009] Christoph Benzmüller, Chad Brown, and Michael Kohlhase. Cut-simulation and impredicativity. *Logical Methods in Computer Science*, 5(1:6):1–21, 2009.

[Benzmüller *et al.*, 2012a] C. Benzmüller, D. Gabbay, V. Genovese, and D. Rispoli. Embedding and automating conditional logics in classical higher-order logic. *Ann. Math. Artif. Intell.*, 66(1-4):257–271, 2012.

[Benzmüller *et al.*, 2012b] C. Benzmüller, J. Otten, and Th. Raths. Implementing and evaluating provers for first-order modal logics. In *Proc. of ECAI 2012*, pp. 163-168, IOS Press, 2012.

[Blanchette and Nipkow, 2010] J.C. Blanchette and T. Nipkow. Nitpick: A counterexample generator for higher-order logic based on a relational model finder. In *Proc. of ITP 2010*, vol. 6172 of *LNCS*, pp. 131–146. Springer, 2010.

[Brown, 2012] C.E. Brown. Satallax: An automated higher-order prover. In *Proc. of IJCAR 2012*, vol. 7364 of LNCS, pp. 111 – 117. Springer, 2012.

[Chellas, 1975] B.F. Chellas. Basic conditional logic. *Journal of Philosophical Logic*, 4(2):133–153, 1975.

[Chellas, 1980] B.F. Chellas. *Modal Logic: An Introduction*. Cambridge: Cambridge University Press, 1980.

[Church, 1940] A. Church. A formulation of the simple theory of types. *Journal of Symbolic Logic*, 5:56–68, 1940.

[Delgrande, 1987] J.P. Delgrande. A first-order conditional logic for prototypical properties. *Artificial Intelligence*, 33(1):105–130, 1987.

[Delgrande, 1998] J.P. Delgrande. On first-order conditional logics. *Artificial Intelligence*, 105(1-2):105–137, 1998.

[Fitting, 2002] M. Fitting. Interpolation for first order S5. *Journal of Symbolic Logic*, 67(2):621–634, 2002.

[Friedman *et al.*, 2000] N. Friedman, J.Y. Halpern, and D. Koller. First-order conditional logic for default reasoning revisited. *ACM Transactions on Computational Logic*, 1(2):175–207, 2000.

[Lehmann and Magidor, 1990] D.J. Lehmann and M. Magidor. Preferential logics: the predicate calculus case. In *Proc' of TARK 1990*, pp. 57–72. Morgan Kaufmann, 1990.

[Lindblad, 2012] F. Lindblad. agsyHol website:. `https://github.com/frelindb/agsyHOL`, 2012.

[Matuszek *et al.*, 2006] C. Matuszek, J. Cabral, M. Witbrock, and J. DeOliveira. An Introduction to the Syntax and Content of Cyc. In *Proc. of the 2006 AAAI Spring Symposium on Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering*, pp. 44–49, 2006.

[Myers *et al.*, 2009] R.S.R. Myers, D. Pattinson, and L. Schröder. Coalgebraic hybrid logic. In *Proc. of FOSSACS 2009*, vol. 5504 of *LNCS*, pp. 137–151. Springer, 2009.

[Niles and Pease, 2001] I. Niles and A. Pease. Towards a Standard Upper Ontology. In *Proc. of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001)*, Ogunquit, Maine, October 17-19 2001.

[Nipkow *et al.*, 2002] T. Nipkow, L.C. Paulson, and M. Wenzel. *Isabelle/HOL: A Proof Assistant for Higher-Order Logic*. Vol. 2283 in LNCS. Springer, 2002.

[Nute, 1980] D. Nute. *Topics in conditional logic*. Reidel, Dordrecht, 1980.

[Olivetti *et al.*, 2007] N. Olivetti, G.L. Pozzato, and C. Schwind. A sequent calculus and a theorem prover for standard conditional logics. *ACM Transactions on Computational Logic*, 8(4), 2007.

[Pease, 2011] A. Pease. *Ontology: A Practical Guide*. Articulate Software Press, Angwin, CA, 2011.

[Pozzato, 2010] G.L. Pozzato. *Conditional and Preferential Logics: Proof Methods and Theorem Proving*, vol. 208 of *Frontiers in Artif. Intell. and Applic.* IOS Press, 2010.

[Schlechta, 1995] K. Schlechta. Defaults as generalized quantifiers. *Journal of Logic and Computation*, 5(4):473–494, 1995.

[Schröder *et al.*, 2010] L. Schröder, D. Pattinson, and D. Hausmann. Optimal tableaux for conditional logics with cautious monotonicity. In *Proc. of ECAI 2010*, pp. 707–712, 2010.

[Stalnaker, 1968] R.C. Stalnaker. A theory of conditionals. In *Studies in Logical Theory*, pp. 98–112. Blackwell, 1968.

[Sutcliffe and Benzmüller, 2010] G. Sutcliffe and C. Benzmüller. Automated reasoning in higher-order logic using the TPTP THF infrastructure. *Journal of Formalized Reasoning*, 3(1):1–27, 2010.

[Sutcliffe, 2009] G. Sutcliffe. The TPTP problem library and associated infrastructure. *Journal of Automated Reasoning*, 43(4):337–362, 2009.