Experimenting with Carmo and Jones' DDL

Lavanya Singh

March 21, 2021

Contents

theory *carmojones-DDL*
imports
Main

begin

Referencing Benzmuller and Parent's implementation [1]

This theory contains the axiomatization of the system and some useful abbreviations.

1 System Definition

1.1 Definitions

This section contains definitions and constants necessary to construct a DDL model.

typedecl i — i is the type for a set of possible worlds."

type-synonym $t = (i \Rightarrow \mathit{bool})$
— t represents a set of DDL formulas.
— this set is defined by its truth function, mapping the set of worlds to the formula set's truth value.

— accessibility relations map a set of worlds to:
consts $av::i \Rightarrow t$ — actual versions of that world set
— these worlds represent what is "open to the agent"
— for example, the agent eating pizza or pasta for dinner might constitute two different actual worlds

consts $pv::i \Rightarrow t$ — possible versions of that world set
— these worlds represent was was "potentially open to the agent"
— for example, what someone across the world eats for dinner might constitute a possible world, — since the agent has no control over this

consts $ob::t \Rightarrow (t \Rightarrow \mathit{bool})$ — set of propositions obligatory in this "context"
— ob(context)(term) is True if t is obligatory in the context

consts $cw::i$ — current world

1.2 Axiomatization

This subsection contains axioms. Because the embedding is semantic, these are just constraints on models.

This axiomatization comes from [2] p6 and the HOL embedding defined in Benzmuller and Parent

axiomatization where
ax-3a: $\forall w.\exists x.\ av(w)(x)$
— every world has some actual version

and *ax-4a*: $\forall w\ x.\ av(w)(x) \longrightarrow pv(w)(x)$
— all actual versions of a world are also possible versions of it

and *ax-4b*: $\forall w.\ pv(w)(w)$
— every world is a possible version of itself

and *ax-5a*: $\forall X.\neg ob(X)(\lambda w.\ False)$
— in any arbitrary context X, something will be obligatory

and *ax-5b*: $\forall X\ Y\ Z.\ (\forall w.\ ((X(w) \wedge Y(w)) \longleftrightarrow (X(w) \wedge Z(w)))) \longrightarrow (ob(X)(Y) \longleftrightarrow ob(X)(Z))$ — note that X(w) denotes w is a member of X
— X, Y, and Z are sets of formulas
— If X ∩ Y = X ∩ Z then the context X obliges Y iff it obliges Z

— ob(X)(λ w. Fw) can be read as F ∈ ob(X)

and *ax-5c2*: $\forall X\ Y\ Z.\ (((\exists w.\ (X(w) \wedge Y(w) \wedge Z(w))) \wedge ob(X)(Y) \wedge ob(X)(Z))) \longrightarrow ob(X)(\lambda w.\ Y(w) \wedge Z(w))$

and *ax-5d*: $\forall X\ Y\ Z.\ ((\forall w.\ Y(w) \longrightarrow X(w)) \wedge ob(X)(Y) \wedge (\forall w.\ X(w) \longrightarrow Z(w)))$

$\longrightarrow ob(Z)(\lambda w.(Z(w) \wedge \neg X(w)) \vee Y(w))$
— If some subset Y of X is in ob(X) then in a larger context Z, any obligatory proposition must either be in Y or in Z-X

and *ax-5e*: $\forall X\ Y\ Z.\ ((\forall w.\ Y(w) \longrightarrow X(w)) \wedge ob(X)(Z) \wedge (\exists w.\ Y(w) \wedge Z(w))) \longrightarrow ob(Y)(Z)$
— If Z is obligatory in context X, then Z is obligatory in a subset of X called Y, if Z shares some elements with Y

1.3 Abbreviations

These abbreviations are defined in @citeBenzmullerParent p9

These are all syntactic sugar for HOL expressions, so evaluating these symbols will be light-weight

— propositional logic symbols
abbreviation *ddlneg*::$t \Rightarrow t$ (¬)
  where $\neg A \equiv \lambda w.\ \neg A(w)$
abbreviation *ddlor*::$t \Rightarrow t \Rightarrow t$ (∨)
  where $\vee\ A\ B \equiv \lambda w.\ (A(w) \vee B(w))$
abbreviation *ddland*::$t \Rightarrow t \Rightarrow t$ (-∧-)
  where $A \wedge\ B \equiv \lambda w.\ (A(w) \wedge B(w))$
abbreviation *ddlif*::$t \Rightarrow t \Rightarrow t$ (-→-)
  where $A \rightarrow B \equiv (\lambda w.\ A(w) \longrightarrow B(w))$

abbreviation *ddlequiv*::$t \Rightarrow t \Rightarrow t$ (-≡-)
  where $(A \equiv B) \equiv ((A \longrightarrow B) \wedge (B \longrightarrow A))$

— modal operators
abbreviation *ddlbox*::$t \Rightarrow t$ (□)
  where $\Box\ A \equiv \lambda w. \forall\, y.\ A(y)$
abbreviation *ddldiamond*::$t \Rightarrow t$ (◊)
  where $\Diamond A \equiv \neg(\Box(\neg A))$

— O{$B|A$} can be read as "B is obligatory in the context A"
abbreviation *ddlob*::$t \Rightarrow t \Rightarrow t$ (O{-|-})
  where $O\{B|A\} \equiv \lambda\ w.\ ob(A)(B)$

— modal symbols over the actual and possible worlds relations
abbreviation *ddlboxa*::$t \Rightarrow t$ ($\Box_a$)
  where $\Box_a A \equiv \lambda x. \forall\, y.\ (\neg\ av(x)(y) \vee A(y))$
abbreviation *ddldiamonda*::$t \Rightarrow t$ ($\Diamond_a$)
  where $\Diamond_a A \equiv \neg(\Box_a(\neg A))$
abbreviation *ddlboxp*::$t \Rightarrow t$ ($\Box_p$)
  where $\Box_p A \equiv \lambda x. \forall\, y.\ (\neg\ pv(x)(y) \vee A(y))$
abbreviation *ddldiamondp*::$t \Rightarrow t$ ($\Diamond_p$)
  where $\Diamond_p A \equiv \neg(\Box_a(\neg A))$

— obligation symbols over the actual and possible worlds
abbreviation *ddloba*::$t \Rightarrow t$ (O_a)
  where $O_a\ A \equiv \lambda x.\ ob(av(x))(A) \wedge (\exists\, y.(av(x)(y) \wedge \neg A(y)))$
abbreviation *ddlobp*::$t \Rightarrow t$ (O_p)
  where $O_p\ A \equiv \lambda x.\ ob(pv(x))(A) \wedge (\exists\, y.(pv(x)(y) \wedge \neg A(y)))$

— syntactic sugar for a "monadic" obligation operator
abbreviation *ddltrue*::t (⊤)
  where $\top \equiv \lambda w.\ True$
abbreviation *ddlob-normal*::$t \Rightarrow t$ (O {-})
  where $(O\ \{A\}) \equiv (O\{A|\top\})$

— validity
abbreviation *ddlvalid*::$t \Rightarrow bool$ (⊨-)
  where $\models A \equiv \forall\, w.\ A\ w$
abbreviation *ddlvalidcw*::$t \Rightarrow bool$ ($\models_c$-)
  where $\models_c A \equiv A\ cw$

1.4 Consistency

Consistency is so easy to show in Isabelle!

lemma *True* **nitpick** [*satisfy*,*user-axioms*,*show-all*,*format=2*] **oops**
— Nitpick successfully found a countermodel.
— It's not shown in the document printout, hence the oops.
— If you hover over "nitpick" in JEdit, the model will be printed to output.

end

theory *carmojones-DDL-completeness* **imports** *carmojones-DDL*

begin

This theory shows completeness for this logic with respect to the models presented in carmojonesDDl.thy.

2 Inference Rules

2.1 Basic Inference Rules

These inference rules are common to most modal and propostional logics

lemma *modus-ponens*: **assumes** $\models A$ **assumes** $\models (A \rightarrow B)$
 shows $\models B$
 using *assms(1) assms(2)* **by** *blast*

— Because I have not defined a "derivable" operator, inference rules are written using assumptions.
— For further meta-logical work, defining metalogical operators may be useful

lemma *nec*: **assumes** $\models A$ **shows** $\models (\Box A)$
 by (*simp add*: *assms*)

lemma *nec-a*: **assumes** $\models A$ **shows** $\models (\Box_a A)$
 by (*simp add*: *assms*)
lemma *nec-p*: **assumes** $\models A$ **shows** $\models (\Box_p A)$
 by (*simp add*: *assms*)

2.2 Fancier Inference Rules

These are new rules that Carmo and Jones introduced for this logic.

lemma *Oa-boxaO*:
 assumes $\models (B \rightarrow ((\neg(\Box((O_a\ A) \rightarrow ((\Box_a w) \wedge O\{A|w\}))))))$
 shows $\models (B \rightarrow (\neg(\Diamond(O_a\ A))))$
 oops
lemma *Oa-boxpO*:
 assumes $\models (B \rightarrow ((\neg(\Box((O_p\ A) \rightarrow ((\Box_p w) \wedge O\{A|w\}))))))$
 shows $\models (B \rightarrow (\neg(\Diamond(O_p\ A))))$
 oops

— The oops indicates that we were not able to find a proof for these lemmas.

B and A must not contain w. not sure how to encode that requirement. one option is to define a new free variables predicate and use that, but that requires a deeper embedding than I have. If Benzmuller and Parent can survive without these inference rules, so can I

3 Axioms

3.1 Box

— $\Box$ is an S5 modal operator, which is where these axioms come from.

lemma *K*:
  shows $\models ((\Box(A \rightarrow B)) \rightarrow ((\Box A) \rightarrow (\Box B)))$
  by *blast*

lemma *T*:
  shows $\models ((\Box A) \rightarrow A)$
  by *blast*

lemma *5*:
  shows $\models ((\Diamond A) \rightarrow (\Box(\Diamond A)))$
  by *blast*

3.2 O

This characterization of O comes from Carmo and Jones p 593

lemma *O-diamond*:
  shows $\models (O\{A|B\} \rightarrow (\Diamond(B \wedge A)))$
  using *ax-5b ax-5a*
  by *metis*
— A is only obligatory in a context if it can possibly be true in that context.

lemma *O-C*:
  shows $\models(((\Diamond(A \wedge (B \wedge C))) \wedge (O\{B|A\} \wedge O\{C|A\})) \rightarrow (O\{B\wedge C|A\}))$
  by (*metis ax-5c2*)
— The conjunction of obligations in a context is obligatory in that context.
— The restriction $\Diamond(ABC)$ is to prevent contradictory obligations and contexts.

lemma *O-SA*:
  shows $\models(((\Box(A \rightarrow B)) \wedge ((\Diamond(A \wedge C)) \wedge O\{C|B\})) \rightarrow (O\{C|A\}))$
  using *ax-5e* **by** *blast*
— The principle of strengthening the antecedent.

lemma *O-REA*:
  shows $\models((\Box(A \equiv B)) \rightarrow (O\{C|A\} \equiv O\{C|B\}))$
  using *O-diamond ax-5e* **by** *blast*
— Equivalence for equivalent contexts.

lemma *O-contextual-REA*:
  shows $\models ((\Box(C \rightarrow (A \equiv B))) \rightarrow (O\{A|C\} \equiv O\{B|C\}))$
  by (*metis ax-5b*)
— The above lemma, but in some context C.

lemma *O-nec*:
  shows $\models(O\{B|A\} \rightarrow (\Box O\{B|A\}))$

by *simp*
— Obligations are necessarily obligated.

lemma *ax-5b''*:
shows $ob\ X\ Y \longleftrightarrow ob\ X\ (\lambda z.\ (Y\ z) \wedge (X\ z))$
by (*metis* (*no-types*, *lifting*) *ax-5b*)

lemma *O-to-O*:
shows $\models(O\{B|A\} \rightarrow O\{(A \rightarrow B)|\top\})$
proof−
have $\forall X\ Y\ Z.\ (ob\ X\ Y \wedge (\forall w.\ X\ w \longrightarrow Z\ w)) \longrightarrow ob\ Z\ (\lambda w.(Z\ w \wedge \neg X\ w) \vee Y\ w)$
by (*smt ax-5d ax-5b ax-5b''*)
thus *?thesis*
proof −
have *f1*: $\forall p\ pa\ pb.\ ((\neg\ (ob\ p\ pa)) \vee (\exists i.\ (p \wedge (\neg\ pb))\ i)) \vee (ob\ pb\ (\vee\ (pb \wedge (\neg\ p))\ pa))$
using ‹$\forall X\ Y\ Z.\ ob\ X\ Y \wedge (\models(X \rightarrow Z)) \longrightarrow ob\ Z\ (\vee\ (Z \wedge (\neg\ X))\ Y)$› **by** *force*
obtain $ii :: (i \Rightarrow bool) \Rightarrow (i \Rightarrow bool) \Rightarrow i$ **where**
$\forall x0\ x2.\ (\exists v3.\ (x2 \wedge (\neg\ x0))\ v3) = (x2 \wedge (\neg\ x0))\ (ii\ x0\ x2)$
by *moura*
then have $\forall p\ pa\ pb.\ ((\neg\ ob\ p\ pa) \vee (p \wedge (\neg\ pb))\ (ii\ pb\ p)) \vee ob\ pb\ (\vee\ (pb \wedge (\neg\ p))\ pa)$
using *f1* **by** *presburger*
then show *?thesis*
by *fastforce*
qed
qed
— Moving from the dyadic to monadic obligation operators.

3.3 Possible Box

— $\Box_p$ is a KT modal operator.
lemma *K-boxp*:
shows $\models((\Box_p(A \rightarrow B)) \rightarrow ((\Box_p A) \rightarrow (\Box_p B)))$
by *blast*
lemma *T-boxp*:
shows $\models((\Box_p A) \rightarrow A)$
using *ax-4b* **by** *blast*

3.4 Actual Box

— $\Box_a$ is a KD modal operator.
lemma *K-boxa*:
shows $\models((\Box_a(A \rightarrow B)) \rightarrow ((\Box_a A) \rightarrow (\Box_a B)))$
by *blast*
lemma *D-boxa*:
shows $\models((\Box_a A) \rightarrow (\Diamond_a A))$
using *ax-3a* **by** *blast*

3.5 Relations Between the Modal Operators

— Relation between $\Box$, $\Box_a$, and $\Box_p$.
lemma *box-boxp*:
  shows $\models((\Box A) \rightarrow (\Box_p A))$
  by *auto*
lemma *boxp-boxa*:
  shows $\models((\Box_p A) \rightarrow (\Box_a A))$
  using *ax-4a* **by** *blast*

— Relation between actual and possible O and $\Box$.
lemma *not-Oa*:
  shows $\models((\Box_a A) \rightarrow ((\neg(O_a\ A)) \wedge (\neg(O_a\ (\neg A)))))$
  using *O-diamond* **by** *blast*
lemma *not-Op*:
shows $\models((\Box_p A) \rightarrow ((\neg(O_p\ A)) \wedge (\neg(O_p\ (\neg A)))))$
  using *O-diamond* **by** *blast*
lemma *equiv-Oa*:
  shows $\models((\Box_a(A \equiv B)) \rightarrow ((O_a\ A) \equiv (O_a\ B)\ ))$
  using *O-contextual-REA* **by** *blast*
lemma *equiv-Op*:
  shows $\models((\Box_p(A \equiv B)) \rightarrow ((O_p\ A) \equiv (O_p\ B)\ ))$
  using *O-contextual-REA* **by** *blast*

— relationships between actual and possible O and $\Box$ and O proper.
lemma *factual-detach-a*:
  shows $\models(((O\{B|A\} \wedge (\Box_a A)) \wedge ((\Diamond_a B) \wedge (\Diamond_a(\neg B)))) \rightarrow (O_a\ B))$
  using *O-SA* **by** *auto*
lemma *factual-detach-p*:
  shows $\models(((O\{B|A\} \wedge (\Box_p A)) \wedge ((\Diamond_p B) \wedge (\Diamond_p(\neg B)))) \rightarrow (O_p\ B))$
  by (*smt O-SA boxp-boxa*)

end

theory *categorical-imperative-1* **imports** *carmojones-DDL-completeness*

begin

4 The Categorical Imperative

4.1 Simple Formulation of the Kingdom of Ends

This is my first attempt at formalizing the concept of the Kingdom of Ends

NOTE: this attempt revealed a bug in my embedding. I've included it as an artifact, but none of these theorems hold anymore (hence the oops).

abbreviation *ddlpermissable*::$t \Rightarrow t$ (P-)
  where $(P\ A) \equiv (\neg(O\ \{\neg A\}))$
— This operator represents permissibility
— Will be useful when discussing the categorical imperative
— Something is permissible if it is not prohibited
— Something is prohibited if its negation is obligatory

lemma *kingdom-of-ends-1*:
  shows $\models ((O\ \{A\}) \rightarrow (\Box\ (P\ A)))$
  oops
— One interpretation of the categorical imperative is that something is obligatory only if it is permissible in every ideal world
— This formulation mirrors the kingdom of ends.
— This formulation is already a theorem of carmo and jones' DDL!.
— It can be shown using the O diamond rule, which just says that obligatory things must be possible.
— There are two possibilities: either the logic is already quite powerful OR this formulation is "empty".

lemma *kingdom-of-ends-2*:
  shows $\models ((\Box\ (P\ A)) \rightarrow (O\ \{A\}))$
  oops

— Notice also that ideally, this relationship does not hold in the reverse direction.
— Plenty of things are necessarily permissible (drinking water) but not obligatory.
— Very strange that this is a theorem in this logic.....
— That being said, Isabelle seems quite upset with this proof and is very slow to resconstruct it
— I am struggling to recreate this proof on paper

lemma *permissible-to-ob*:
  shows $\models ((P\ A) \rightarrow (O\ \{A\}))$
  oops

— Uh-oh.....this shouldn't be true...
— Not all permissable things are obligatory.....

lemma *weaker-permissible-to-ob*:
shows $\models ((\Diamond\ (P\ A)) \rightarrow O\ \{A\})$
oops
— Makes sense that this follows from the reverse kingdom of ends.
— Obligation and necessity/possibility are separated in this logic
— Both the dyadic obligation and necessity operator are world agnostic

lemma *contradictory-obligations*:
shows $\models(\neg\ ((O\ \{A\}) \wedge (O\ \{\neg\ A\})))$
oops
— What is the cause of the above strangeness?
— This very intuitive theorem holds in my logic but not in Benzmuller Parent's
— It's clear that this theorem results in the strange results above.
— Conclusion: There is a bug in my embedding

Sidebar: the above theorem is really intuitive - it seems like we wouldn't want contradictory things to be obligatory in any logic. But for some reason, not only is it not a theorem of Carmo and Jones' logic, it actually implies some strange conclusions, including that everything is either permissible or obligatory. It's not clear to me from a semantic perspective why this would be the case. In fact this theorem seems like a desirable property. Potential avenue for exploration

Did some debugging. What was the problem? A misplaced parentheses in the definition of ax5b that led to a term being on the wrong side of an implication. Computer Science :(

After the debugging, all of this is no longer true! On to the next attempt :)

end

theory *categorical-imperative-naive* **imports** *carmojones-DDL-completeness*

begin

5 The Categorical Imperative

5.1 Simple Formulation of the Formula of Universal Law

This is my second attempt at formalizing the Formula of Universal Law

abbreviation *ddlpermissable*::$t \Rightarrow t$ (P-)
  where $(P\ A) \equiv (\neg(O\ \{\neg A\}))$
— This operator represents permissibility
— Will be useful when discussing the categorical imperative
— Something is permissible if it is not prohibited
— Something is prohibited if its negation is obligatory

Let's consider a naive reading of the Formula of Universal Law (FUL). From the Groundwork, 'act only in accordance with that maxim through which you can at the same time will that it become a universal law'. What does this mean in DDL? One interpretation is if A is not necessarily permissible, then its negation is obligated.

axiomatization where
FUL-1: $\models ((\neg(\Box\ (P\ A))) \rightarrow (O\ \{(\neg A)\}))$

5.2 Basic Tests

lemma *True* **nitpick** [*satisfy*,*user-axioms*,*format=2*] **oops**
— "Nitpick found a model for card i = 1:
Empty assignment"
— Nitpick tells us that the FUL is consistent
— "oops" after Nitpick does not mean Nitpick failed.

lemma *something-is-obligatory*:
  shows $\forall\ w.\ \exists\ A.\ O\ \{A\}\ w$
  nitpick [*user-axioms*]
  oops
— We might think that in every world we want something to be obligated.
— Sadly, Sledgehammer times out trying to prove this. Let's relax this
— "Nitpick found a counterexample for card i = 1:
Empty assignment"
— Nitpick to the rescue! The formula is in fact not valid.

lemma *something-is-obligatory-2*:
  shows $\forall\ w.\ \exists\ A.\ O\ \{A\}\ w$
  nitpick [*user-axioms*, *falsify=false*]
  oops
— "Nitpick found a model for card i = 1:
Skolem constant: A = $(\lambda x.\_)(i_1 := \text{True})$"

— Nitpick tells us that the formula is consistent - it found a model where the formula is true.
— This means that our model is underspecified - this formula is neither valid nor inconsistent.

lemma *something-is-obligatory-relaxed*:
  shows $\exists\ A\ w.\ O\ \{A\}\ w$
  nitpick [*user-axioms*]
  oops
— "Nitpick found a counterexample for card i = 1:
Empty assignment"
— The relaxed version definitely isn't valid.

lemma *something-is-obligatory-relaxed-2*:
  shows $\exists\ A\ w.\ O\ \{A\}\ w$
  nitpick [*user-axioms*, *falsify=false*]
  oops
— "Nitpick found a model for card i = 1:
Skolem constant: A = $(\lambda x.\_)(i_1 :=$ True)"
— Nitpick tells us that the formula is consistent - it found a model where the formula is true.
— The model seems underspecified.

5.3 Specifying the Model

Let's specify the model. What if we add something impermissible?

consts M::t
abbreviation *murder-wrong*::*bool* **where** *murder-wrong* $\equiv \models(O\ \{\neg\ M\})$

lemma *something-is-obligatory-2*:
  assumes *murder-wrong*
  shows $\forall\ w.\ \exists\ A.\ O\ \{A\}\ w$
  using *assms* **by** *auto*
— It works this time, but I think "murder wrong" might be too strong of an assumption

Let's try a weaker assumption: Not everyone can lie.

typedecl *person*
consts *lies*::*person*$\Rightarrow$*t*
consts *me*::*person*

lemma *breaking-promises*:
  assumes $\neg\ (\forall x.\ lie(x)\ cw) \wedge (lie(me)\ cw)$
  shows $(O\ \{\neg\ (lie(me))\})\ cw$
  nitpick [*user-axioms*]
  oops
— No proof found. Makes sense:
— This version of FUL simply universalizes across worlds (using the $\Box$ operator),

— But NOT across people, which is really what the most obvious reading of FUL implies
— "Nitpick found a counterexample for card person = 2 and card i = 2:
Free variable: lie = $(\lambda x.\_)(p_1 := (\lambda x.\_)(i_1 :=$ True, $i_2 :=$ False), $p_2 := (\lambda x.\_)(i_1 :=$ False, $i_2 :=$ False))"

5.4 Consistent Sentences

The above section tested validity. We might also be interested in some weaker properties

Let's test whether certain sentences are consistent - can we find a model that makes them true?

lemma *permissible*:
  fixes A
  shows $((\neg\ (O\ \{A\}))\ \wedge\ (\neg\ (O\ \{\neg\ A\})))\ w$
  nitpick [*user-axioms*, *falsify=false*] **oops**
— "Nitpick found a model for card i = 1:
Free variable: A = $(\lambda x.\_)(i_1 :=$ False)"
— Awesome! Permissible things are consistent - clearly we've fixed the bug from categorical_imperative_1

lemma *conflicting-obligations*:
  fixes A
  shows $(O\ \{A\}\ \wedge\ O\ \{\neg\ A\})\ w$
  nitpick [*user-axioms*, *falsify=false*] **oops**
— "Nitpick found a model for card i = 2:
Free variable: A = $(\lambda x.\_)(i_1 :=$ False, $i_2 :=$ True)"
— Oh no! Nitpick found a model with conflicting obligations - that's bad!

5.5 Metaethical Tests

lemma *FUL-alternate*:
  shows $\models ((\Diamond\ (O\ \{\neg\ A\}))\ \rightarrow\ (O\ \{\neg\ A\}))$
  by *simp*
— One problem becomes obvious if we look at the definition of permissible
— Expanding the FUL gives us: $\sim \Box \sim O(\sim A) \longrightarrow O(\sim A)$
— By modal duals we get that $\diamond O(\sim A) \longrightarrow O(\sim A)$
— This means that if something is possibly prohibited, it is in fact prohibited.
— I'm not convinced that this is a desirable property of an ethical theory.

lemma *arbitrary-obligations*:
  fixes $A{::}t$
  shows $O\ \{A\}\ w$
  nitpick [*user-axioms=true*] **oops**
— "Nitpick found a counterexample for card i = 1:
Free variable: A = $(\lambda x.\_)(i_1 :=$ False)"
— This is good! Shows us that any arbitrary term isn't obligatory.

lemma *removing-conflicting-obligations*:
  assumes $\forall A. \models (\neg (O \{A\} \wedge O \{\neg A\}))$
  shows *True*
  nitpick [*satisfy*,*user-axioms*,*format=2*] **oops**
— " Nitpick found a model for card i = 1:
Empty assignment"
— We can disallow conflicting obligations and the system is still consistent - that's good.

lemma *implied-contradiction*:
  fixes $A::t$
  fixes $B::t$
  assumes $\models(\neg (A \wedge B))$
  shows $\models(\neg (O \{A\} \wedge O \{B\}))$
  nitpick [*user-axioms*]
proof –
  have $\models(\neg(\Diamond(A \wedge B)))$
    by (*simp add*: *assms*)
  then have $\models(\neg (O \{A \wedge B\}))$ **by** (*smt carmojones-DDL-completeness.O-diamond*)
  thus *?thesis* **oops**
— [3] mentions that if two maxims imply a contradiction, they must not be willed.
— Above is a natural interpretation of this fact that we are, so far, unable to prove.
— "Nitpick found a counterexample for card i = 2:
Free variables: A = $(\lambda x.\_)(i_1$:= True, i_2 := False) B = $(\lambda x.\_)(i_1$:= False, i_2 := True)"
— This isn't actually a theorem of the logic as formed - clearly this is a problem.

lemma *distribute-obligations-if*:
  assumes $\models O \{A \wedge B\}$
  shows $\models (O \{A\} \wedge O \{B\})$
  nitpick [*user-axioms*, *falsify=true*, *verbose*]
  oops
— Nitpick can't find a countermodel for this theorem, and sledgehammer can't find a proof.
— Super strange. I wonder if this is similar to $\Box(A \wedge B)$ vs $\Box A \wedge \Box B$

lemma *distribute-obligations-onlyif*:
  assumes $\models (O \{A\} \wedge O \{B\})$
  shows $\models O \{A \wedge B\}$
  nitpick [*user-axioms*] **oops**
— "Nitpick found a counterexample for card i = 2:
Free variables: A = $(\lambda x.\_)(i_1$:= True, i_2 := False) B = $(\lambda x.\_)(i_1$:= False, i_2 := True)"
— If this was a theorem, then contradictory obligations would be ruled out pretty immediately.
— Note that all of this holds in CJ's original DDL as well, not just my modified version.
— We might imagine adding this equivalence to our system.
end

References

[1] C. Benzmüller, A. Farjami, and X. Parent. Faithful semantical embedding of a dyadic deontic logic in HOL. *CoRR*, abs/1802.08454, 2018.

[2] J. Carmo and A. Jones. Completeness and decidability results for a logic of contrary-to-duty conditionals. *J. Log. Comput.*, 23:585–626, 2013.

[3] C. Korsgaard. Kant's Formula of Universal Law. *Pacific Philosophical Quarterly*, 66:24–47, 1985.