Formalisation and Evaluation of Alan Gewirth's Proof for the Principle of Generic Consistency in Isabelle/HOL*

David Fuenmayor[1] and Christoph Benzmüller[1,2]

[1]Freie Universität Berlin, Germany
[2]University of Luxembourg, Luxembourg

March 17, 2025

Abstract

An ambitious ethical theory —Alan Gewirth's "Principle of Generic Consistency"— is encoded and analysed in Isabelle/HOL. Gewirth's theory has stirred much attention in philosophy and ethics and has been proposed as a potential means to bound the impact of artificial general intelligence.

Contents

*Benzmüller received support from the Volkswagen Foundation (Project CRAP: Consistent Rational Argumentation in Politics).

1 Introduction

We present an encoding of an ambitious ethical theory —Alan Gewirth's "Principle of Generic Consistency (PGC)"— in Isabelle/HOL. The PGC has stirred much attention in philosophy and ethics [4] and has been proposed as a potential means to bound the impact of artificial general intelligence (AGI) [9]. With our contribution we make a first, important step towards formally assessing the PGC and its potential applications in AI. Our formalisation utilises the shallow semantical embedding approach [3] and adapts a recent embedding of dyadic deontic logic in HOL [1] [2].

2 Semantic Embedding of Carmo and Jones' Dyadic Deontic Logic (DDL) augmented with Kaplanian contexts

We introduce a modification of the semantic embedding developed by Benzmüller et al. [1] [2] for the Dyadic Deontic Logic originally presented by Carmo and Jones [5]. We extend this embedding to a two-dimensional semantics as originally presented by David Kaplan [7] [8].

2.1 Definition of Types

typedecl w — Type for possible worlds (Kaplan's "circumstances of evaluation" or "counterfactual situations")
typedecl e — Type for individuals (entities eligible to become agents)
typedecl c — Type for Kaplanian "contexts of use"
type-synonym $wo = w \Rightarrow bool$ — contents/propositions are identified with their truth-sets
type-synonym $cwo = c \Rightarrow wo$ — sentence meaning (Kaplan's "character") is a function from contexts to contents
type-synonym $m = cwo$ — we use the letter 'm' for characters (reminiscent of "meaning")

2.2 Semantic Characterisation of DDL

2.2.1 Basic Set Operations

abbreviation *subset*::*wo*⇒*wo*⇒*bool* (**infix** ‹$\sqsubseteq$› *46*) **where** $\alpha \sqsubseteq \beta \equiv \forall w.\ \alpha\ w \longrightarrow \beta\ w$
abbreviation *intersection*::*wo*⇒*wo*⇒*wo* (**infixr** ‹$\sqcap$› *48*) **where** $\alpha \sqcap \beta \equiv \lambda x.\ \alpha\ x \wedge \beta\ x$
abbreviation *union*::*wo*⇒*wo*⇒*wo* (**infixr** ‹$\sqcup$› *48*) **where** $\alpha \sqcup \beta \equiv \lambda x.\ \alpha\ x \vee \beta\ x$
abbreviation *complement*::*wo*⇒*wo* (‹∼-›[*45*]*46*) **where** $\sim\alpha \equiv \lambda x.\ \neg\alpha\ x$
abbreviation *instantiated*::*wo*⇒*bool* (‹$\mathcal{I}$-›[*45*]*46*) **where** $\mathcal{I}\ \varphi \equiv \exists x.\ \varphi\ x$
abbreviation *setEq*::*wo*⇒*wo*⇒*bool* (**infix** ‹$=_s$› *46*) **where** $\alpha =_s \beta \equiv \forall x.\ \alpha\ x \longleftrightarrow \beta\ x$
abbreviation *univSet* :: *wo* (‹$\top$›) **where** $\top \equiv \lambda w.\ True$
abbreviation *emptySet* :: *wo* (‹$\bot$›) **where** $\bot \equiv \lambda w.\ False$

2.2.2 Set-Theoretic Conditions for DDL

consts
av::*w*⇒*wo* — set of worlds that are open alternatives (aka. actual versions) of w
pv::*w*⇒*wo* — set of worlds that are possible alternatives (aka. potential versions) of w
ob::*wo*⇒*wo*⇒*bool* — set of propositions which are obligatory in a given context (of type wo)

axiomatization where
sem-3a: $\forall w.\ \mathcal{I}(av\ w)$ **and** — av is serial: in every situation there is always an open alternative
sem-4a: $\forall w.\ av\ w \sqsubseteq pv\ w$ **and** — open alternatives are possible alternatives
sem-4b: $\forall w.\ pv\ w\ w$ **and** — pv is reflexive: every situation is a possible alternative to itself
sem-5a: $\forall X.\ \neg(ob\ X\ \bot)$ **and** — contradictions cannot be obligatory
sem-5b: $\forall X\ Y\ Z.\ (X \sqcap Y) =_s (X \sqcap Z) \longrightarrow (ob\ X\ Y \longleftrightarrow ob\ X\ Z)$ **and**
sem-5c: $\forall X\ Y\ Z.\ \mathcal{I}(X \sqcap Y \sqcap Z) \wedge ob\ X\ Y \wedge ob\ X\ Z \longrightarrow ob\ X\ (Y \sqcap Z)$ **and**
sem-5d: $\forall X\ Y\ Z.\ (Y \sqsubseteq X \wedge ob\ X\ Y \wedge X \sqsubseteq Z) \longrightarrow ob\ Z\ ((Z \sqcap (\sim X)) \sqcup Y)$ **and**
sem-5e: $\forall X\ Y\ Z.\ Y \sqsubseteq X \wedge ob\ X\ Z \wedge \mathcal{I}(Y \sqcap Z) \longrightarrow ob\ Y\ Z$

lemma *True* **nitpick**[*satisfy*] **oops** — model found: axioms are consistent

2.2.3 Verifying Semantic Conditions

lemma *sem-5b1*: $ob\ X\ Y \longrightarrow ob\ X\ (Y \sqcap X)$ **by** (*metis* (*no-types*, *lifting*) *sem-5b*)
lemma *sem-5b2*: $(ob\ X\ (Y \sqcap X) \longrightarrow ob\ X\ Y)$ **by** (*metis* (*no-types*, *lifting*) *sem-5b*)
lemma *sem-5ab*: $ob\ X\ Y \longrightarrow \mathcal{I}(X \sqcap Y)$ **by** (*metis* (*full-types*) *sem-5a sem-5b*)
lemma *sem-5bd1*: $Y \sqsubseteq X \wedge ob\ X\ Y \wedge X \sqsubseteq Z \longrightarrow ob\ Z\ ((\sim X) \sqcup Y)$ **using** *sem-5b sem-5d* **by** *smt*
lemma *sem-5bd2*: $ob\ X\ Y \wedge X \sqsubseteq Z \longrightarrow ob\ Z\ ((Z \sqcap (\sim X)) \sqcup Y)$ **using** *sem-5b sem-5d* **by** (*smt sem-5b1*)
lemma *sem-5bd3*: $ob\ X\ Y \wedge X \sqsubseteq Z \longrightarrow ob\ Z\ ((\sim X) \sqcup Y)$ **by** (*smt sem-5bd2 sem-5b*)
lemma *sem-5bd4*: $ob\ X\ Y \wedge X \sqsubseteq Z \longrightarrow ob\ Z\ ((\sim X) \sqcup (X \sqcap Y))$ **using** *sem-5bd3* **by** *auto*
lemma *sem-5bcd*: $(ob\ X\ Z \wedge ob\ Y\ Z) \longrightarrow ob\ (X \sqcup Y)\ Z$ **using** *sem-5b sem-5c sem-5d* **oops**

lemma $ob\ A\ B \longleftrightarrow (\mathcal{I}(A \sqcap B) \wedge (\forall X.\ X \sqsubseteq A \wedge \mathcal{I}(X \sqcap B) \longrightarrow ob\ X\ B))$ **using** *sem-5e sem-5ab* **by** *blast*

2.3 (Shallow) Semantic Embedding of DDL

2.3.1 Basic Propositional Logic

abbreviation *pand*::*m*⇒*m*⇒*m* (**infixr**‹∧› *51*) **where** $\varphi{\wedge}\psi \equiv \lambda c\ w.\ (\varphi\ c\ w){\wedge}(\psi\ c\ w)$
abbreviation *por*::*m*⇒*m*⇒*m* (**infixr**‹∨› *50*) **where** $\varphi{\vee}\psi \equiv \lambda c\ w.\ (\varphi\ c\ w){\vee}(\psi\ c\ w)$

abbreviation *pimp*::$m \Rightarrow m \Rightarrow m$ (**infix**‹→› *49*) **where** $\varphi \rightarrow \psi \equiv \lambda c\ w.\ (\varphi\ c\ w) \longrightarrow (\psi\ c\ w)$
abbreviation *pequ*::$m \Rightarrow m \Rightarrow m$ (**infix**‹↔› *48*) **where** $\varphi \leftrightarrow \psi \equiv \lambda c\ w.\ (\varphi\ c\ w) \longleftrightarrow (\psi\ c\ w)$
abbreviation *pnot*::$m \Rightarrow m$ (‹¬-› *[52]53*) **where** $\neg\varphi \equiv \lambda c\ w.\ \neg(\varphi\ c\ w)$

2.3.2 Modal Operators

abbreviation *cjboxa* :: $m \Rightarrow m$ (‹$\Box_a$-› *[52]53*) **where** $\Box_a\varphi \equiv \lambda c\ w.\ \forall v.\ (av\ w)\ v \longrightarrow (\varphi\ c\ v)$
abbreviation *cjdiaa* :: $m \Rightarrow m$ (‹$\Diamond_a$-› *[52]53*) **where** $\Diamond_a\varphi \equiv \lambda c\ w.\ \exists v.\ (av\ w)\ v \wedge (\varphi\ c\ v)$
abbreviation *cjboxp* :: $m \Rightarrow m$ (‹$\Box_p$-› *[52]53*) **where** $\Box_p\varphi \equiv \lambda c\ w.\ \forall v.\ (pv\ w)\ v \longrightarrow (\varphi\ c\ v)$
abbreviation *cjdiap* :: $m \Rightarrow m$ (‹$\Diamond_p$-› *[52]53*) **where** $\Diamond_p\varphi \equiv \lambda c\ w.\ \exists v.\ (pv\ w)\ v \wedge (\varphi\ c\ v)$
abbreviation *cjtaut* :: m (‹⊤›) **where** $\top \equiv \lambda c\ w.\ True$
abbreviation *cjcontr* :: m (‹⊥›) **where** $\bot \equiv \lambda c\ w.\ False$

2.3.3 Deontic Operators

abbreviation *cjod* :: $m \Rightarrow m \Rightarrow m$ (‹$\mathbf{O}\langle$-|-$\rangle$›*54*) **where** $\mathbf{O}\langle\varphi|\sigma\rangle \equiv \lambda c\ w.\ ob\ (\sigma\ c)\ (\varphi\ c)$
abbreviation *cjoa* :: $m \Rightarrow m$ (‹$\mathbf{O}_a$-› *[53]54*) **where** $\mathbf{O}_a\varphi \equiv \lambda c\ w.\ (ob\ (av\ w))\ (\varphi\ c) \wedge (\exists x.\ (av\ w)\ x \wedge \neg(\varphi\ c\ x))$
abbreviation *cjop* :: $m \Rightarrow m$ (‹$\mathbf{O}_i$-› *[53]54*) **where** $\mathbf{O}_i\varphi \equiv \lambda c\ w.\ (ob\ (pv\ w))\ (\varphi\ c) \wedge (\exists x.\ (pv\ w)\ x \wedge \neg(\varphi\ c\ x))$

2.3.4 Logical Validity (Classical)

abbreviation *modvalidctx* :: $m \Rightarrow c \Rightarrow bool$ (‹$\lfloor$-$\rfloor^M$›) **where** $\lfloor\varphi\rfloor^M \equiv \lambda c.\ \forall w.\ \varphi\ c\ w$ — context-dependent modal validity
abbreviation *modvalid* :: $m \Rightarrow bool$ (‹$\lfloor$-$\rfloor$›) **where** $\lfloor\varphi\rfloor \equiv \forall c.\ \lfloor\varphi\rfloor^M\ c$ — general modal validity (modally valid in each context)

2.4 Verifying the Embedding

2.4.1 Avoiding Modal Collapse

lemma $\lfloor P \rightarrow \mathbf{O}_a P\rfloor$ **nitpick oops** — (actual) deontic modal collapse is countersatisfiable
lemma $\lfloor P \rightarrow \mathbf{O}_i P\rfloor$ **nitpick oops** — (ideal) deontic modal collapse is countersatisfiable
lemma $\lfloor P \rightarrow \Box_a P\rfloor$ **nitpick oops** — alethic modal collapse is countersatisfiable (implies all other necessity operators)

2.4.2 Necessitation Rule

lemma *NecDDLa*: $\lfloor A\rfloor \Longrightarrow \lfloor\Box_a A\rfloor$ **by** *simp*
lemma *NecDDLp*: $\lfloor A\rfloor \Longrightarrow \lfloor\Box_p A\rfloor$ **by** *simp*

2.4.3 Lemmas for Semantic Conditions

abbreviation *mboxS5* :: $m \Rightarrow m$ (‹$\Box^{S5}$-› *[52]53*) **where** $\Box^{S5}\varphi \equiv \lambda c\ w.\ \forall v.\ \varphi\ c\ v$
abbreviation *mdiaS5* :: $m \Rightarrow m$ (‹$\Diamond^{S5}$-› *[52]53*) **where** $\Diamond^{S5}\varphi \equiv \lambda c\ w.\ \exists v.\ \varphi\ c\ v$

lemma *C-2*: $\lfloor\mathbf{O}\langle A \mid B\rangle \rightarrow \Diamond^{S5}(B \wedge A)\rfloor$ **by** (*simp add*: *sem-5ab*)
lemma *C-3*: $\lfloor((\Diamond^{S5}(A \wedge B \wedge C)) \wedge \mathbf{O}\langle B|A\rangle \wedge \mathbf{O}\langle C|A\rangle) \rightarrow \mathbf{O}\langle(B \wedge C)|\ A\rangle\rfloor$ **by** (*simp add*: *sem-5c*)
lemma *C-4*: $\lfloor(\Box^{S5}(A \rightarrow B) \wedge \Diamond^{S5}(A \wedge C) \wedge \mathbf{O}\langle C|B\rangle) \rightarrow \mathbf{O}\langle C|A\rangle\rfloor$ **using** *sem-5e* **by** *blast*
lemma *C-5*: $\lfloor\Box^{S5}(A \leftrightarrow B) \rightarrow (\mathbf{O}\langle C|A\rangle \rightarrow \mathbf{O}\langle C|B\rangle)\rfloor$ **using** *C-2 sem-5e* **by** *blast*
lemma *C-6*: $\lfloor\Box^{S5}(C \rightarrow (A \leftrightarrow B)) \rightarrow (\mathbf{O}\langle A|C\rangle \leftrightarrow \mathbf{O}\langle B|C\rangle)\rfloor$ **by** (*metis sem-5b*)
lemma *C-7*: $\lfloor\mathbf{O}\langle B|A\rangle \rightarrow \Box^{S5}\mathbf{O}\langle B|A\rangle\rfloor$ **by** *blast*
lemma *C-8*: $\lfloor\mathbf{O}\langle B|A\rangle \rightarrow \mathbf{O}\langle A \rightarrow B|\ \top\rangle\rfloor$ **using** *sem-5bd4* **by** *presburger*

2.4.4 Verifying Axiomatic Characterisation

The following theorems have been taken from the original Carmo and Jones' paper ([5] p.293ff).

lemma *CJ-3*: $\lfloor \Box_p A \rightarrow \Box_a A \rfloor$ **by** (*simp add*: *sem-4a*)
lemma *CJ-4*: $\lfloor \neg \mathbf{O}\langle \bot | A \rangle \rfloor$ **by** (*simp add*: *sem-5a*)

lemma *CJ-5*: $\lfloor (\mathbf{O}\langle B|A\rangle \wedge \mathbf{O}\langle C|A\rangle) \rightarrow \mathbf{O}\langle B \wedge C|A\rangle \rfloor$ **nitpick oops** — countermodel found
lemma *CJ-5-minus*: $\lfloor \Diamond^{S5}(A \wedge B \wedge C) \wedge (\mathbf{O}\langle B|A\rangle \wedge \mathbf{O}\langle C|A\rangle) \rightarrow \mathbf{O}\langle B \wedge C|A\rangle \rfloor$ **by** (*simp add*: *sem-5c*)

lemma *CJ-6*: $\lfloor \mathbf{O}\langle B|A\rangle \rightarrow \mathbf{O}\langle B|A \wedge B\rangle \rfloor$ **by** (*smt C-2 C-4*)
lemma *CJ-7*: $\lfloor A \leftrightarrow B \rfloor \longrightarrow \lfloor \mathbf{O}\langle C|A\rangle \leftrightarrow \mathbf{O}\langle C|B\rangle \rfloor$ **using** *sem-5ab sem-5e* **by** *blast*
lemma *CJ-8*: $\lfloor C \rightarrow (A \leftrightarrow B) \rfloor \longrightarrow \lfloor \mathbf{O}\langle A|C\rangle \leftrightarrow \mathbf{O}\langle B|C\rangle \rfloor$ **using** *C-6* **by** *simp*

lemma *CJ-9a*: $\lfloor \Diamond_p \mathbf{O}\langle B|A\rangle \rightarrow \Box_p \mathbf{O}\langle B|A\rangle \rfloor$ **by** *simp*
lemma *CJ-9p*: $\lfloor \Diamond_a \mathbf{O}\langle B|A\rangle \rightarrow \Box_a \mathbf{O}\langle B|A\rangle \rfloor$ **by** *simp*
lemma *CJ-9-var-a*: $\lfloor \mathbf{O}\langle B|A\rangle \rightarrow \Box_a \mathbf{O}\langle B|A\rangle \rfloor$ **by** *simp*
lemma *CJ-9-var-b*: $\lfloor \mathbf{O}\langle B|A\rangle \rightarrow \Box_p \mathbf{O}\langle B|A\rangle \rfloor$ **by** *simp*
lemma *CJ-10*: $\lfloor \Diamond_p(A \wedge B \wedge C) \wedge \mathbf{O}\langle C|B\rangle \rightarrow \mathbf{O}\langle C|A \wedge B\rangle \rfloor$ **by** (*smt C-4*)

lemma *CJ-11a*: $\lfloor (\mathbf{O}_a A \wedge \mathbf{O}_a B) \rightarrow \mathbf{O}_a(A \wedge B) \rfloor$ **nitpick oops** — countermodel found
lemma *CJ-11a-var*: $\lfloor \Diamond_a(A \wedge B) \wedge (\mathbf{O}_a A \wedge \mathbf{O}_a B) \rightarrow \mathbf{O}_a(A \wedge B) \rfloor$ **using** *sem-5c* **by** *auto*

lemma *CJ-11p*: $\lfloor (\mathbf{O}_i A \wedge \mathbf{O}_i B) \rightarrow \mathbf{O}_i(A \wedge B) \rfloor$ **nitpick oops** — countermodel found
lemma *CJ-11p-var*: $\lfloor \Diamond_p(A \wedge B) \wedge (\mathbf{O}_i A \wedge \mathbf{O}_i B) \rightarrow \mathbf{O}_i(A \wedge B) \rfloor$ **using** *sem-5c* **by** *auto*

lemma *CJ-12a*: $\lfloor \Box_a A \rightarrow (\neg \mathbf{O}_a A \wedge \neg \mathbf{O}_a(\neg A)) \rfloor$ **using** *sem-5ab* **by** *blast*
lemma *CJ-12p*: $\lfloor \Box_p A \rightarrow (\neg \mathbf{O}_i A \wedge \neg \mathbf{O}_i(\neg A)) \rfloor$ **using** *sem-5ab* **by** *blast*

lemma *CJ-13a*: $\lfloor \Box_a(A \leftrightarrow B) \rightarrow (\mathbf{O}_a A \leftrightarrow \mathbf{O}_a B) \rfloor$ **using** *sem-5b* **by** *metis*
lemma *CJ-13p*: $\lfloor \Box_p(A \leftrightarrow B) \rightarrow (\mathbf{O}_i A \leftrightarrow \mathbf{O}_i B) \rfloor$ **using** *sem-5b* **by** *metis*

lemma *CJ-O-O*: $\lfloor \mathbf{O}\langle B|A\rangle \rightarrow \mathbf{O}\langle A \rightarrow B|\top\rangle \rfloor$ **using** *sem-5bd4* **by** *presburger*

An ideal obligation which is actually possible both to fulfill and to violate entails an actual obligation ([5] p.319).

lemma *CJ-Oi-Oa*: $\lfloor (\mathbf{O}_i A \wedge \Diamond_a A \wedge \Diamond_a(\neg A)) \rightarrow \mathbf{O}_a A \rfloor$ **using** *sem-5e sem-4a* **by** *blast*

Bridge relations between conditional obligations and actual/ideal obligations:

lemma *CJ-14a*: $\lfloor \mathbf{O}\langle B|A\rangle \wedge \Box_a A \wedge \Diamond_a B \wedge \Diamond_a \neg B \rightarrow \mathbf{O}_a B \rfloor$ **using** *sem-5e* **by** *blast*
lemma *CJ-14p*: $\lfloor \mathbf{O}\langle B|A\rangle \wedge \Box_p A \wedge \Diamond_p B \wedge \Diamond_p \neg B \rightarrow \mathbf{O}_i B \rfloor$ **using** *sem-5e* **by** *blast*

lemma *CJ-15a*: $\lfloor (\mathbf{O}\langle B|A\rangle \wedge \Diamond_a(A \wedge B) \wedge \Diamond_a(A \wedge \neg B)) \rightarrow \mathbf{O}_a(A \rightarrow B) \rfloor$ **using** *CJ-O-O sem-5e* **by** *fastforce*
lemma *CJ-15p*: $\lfloor (\mathbf{O}\langle B|A\rangle \wedge \Diamond_p(A \wedge B) \wedge \Diamond_p(A \wedge \neg B)) \rightarrow \mathbf{O}_i(A \rightarrow B) \rfloor$ **using** *CJ-O-O sem-5e* **by** *fastforce*

3 Extending the Carmo and Jones DDL Logical Framework

In the last section, we have modelled Kaplanian contexts by introducing a new type of object (type c) and modelled sentence meanings as so-called "characters", i.e. functions from contexts to sets of worlds (type $c \Rightarrow w \Rightarrow o$). We also made the corresponding adjustments to the original semantic embedding of Carmo and Jones' DDL [1] [2]. So far we haven't said much about what these Kaplanian contexts are or which effect they should have on the evaluation of logical validity. We restricted ourselves to illustrating that their introduction does not have any influence on the (classical) modal validity of several DDL key theorems. In this section we introduce an alternative notion of logical validity suited for working with contexts: indexical validity [7] [8].

3.1 Context Features

Kaplan's theory ("Logic of Demonstratives" [7]) aims at modelling the behaviour of certain context-sensitive linguistic expressions like the pronouns 'I', 'my', 'you', 'he', 'his', 'she', 'it', the demonstrative pronouns 'that', 'this', the adverbs 'here', 'now', 'tomorrow', 'yesterday', the adjectives 'actual', 'present', and others. Such expressions are known as "indexicals" and so Kaplan's logical system (among others) is usually referred to as a "logic of indexicals" (although in his seminal work he referred to it as a "logic of demonstratives" (LD)) [7]. In the following we will refer to Kaplan's logic as the logic "LD". It is characteristic of an indexical that its content varies with context, i.e. they have a context-sensitive character. Non-indexicals have a fixed character. The same content is invoked in all contexts. Kaplan's logical system models context-sensitivity by representing contexts as tuples of features ($\langle Agent(c),\ Position(c),\ World(c),\ Time(c)\rangle$). The agent and the position of context c can be seen as the actual speaker and place of the utterance respectively, while c's world and time stand for the circumstances of evaluation of the expression's content and allow for the interaction of indexicals with alethic and tense modalities respectively.

To keep things simple (and relevant for our task) we restrict ourselves to representing a context c as the pair: $\langle Agent(c),\ World(c)\rangle$. For this purpose we represent the functional concepts "Agent" and "World" as logical constants.

consts $Agent{::}c \Rightarrow e$ — function retrieving the agent corresponding to context c
consts $World{::}c \Rightarrow w$ — function retrieving the world corresponding to context c

3.2 Logical Validity

Kaplan's notion of (context-dependent) logical truth for a sentence corresponds to its (context-sensitive) formula (of type $c \Rightarrow w \Rightarrow bool$ i.e. m) being true in the given context and at its corresponding world.

abbreviation $ldtruectx{::}m \Rightarrow c \Rightarrow bool$ ($\langle\lfloor\text{-}\rfloor_{\text{-}}\rangle$) **where** $\lfloor\varphi\rfloor_c \equiv \varphi\ c\ (World\ c)$ — truth in the given context

Kaplan's LD notion of logical validity for a sentence corresponds to its being true in all contexts. This notion is also known as indexical validity.

abbreviation $ldvalid{::}m \Rightarrow bool$ ($\langle\lfloor\text{-}\rfloor^D\rangle$) **where** $\lfloor\varphi\rfloor^D \equiv \forall\, c.\ \lfloor\varphi\rfloor_c$ — LD validity (true in every context)

Here we show that indexical validity is indeed weaker than its classical modal counterpart (truth at all worlds for all contexts):

lemma $\lfloor A \rfloor \Longrightarrow \lfloor A \rfloor^D$ **by** *simp*
lemma $\lfloor A \rfloor^D \Longrightarrow \lfloor A \rfloor$ **nitpick oops** — countermodel found

Here we show that the interplay between indexical validity and the DDL modal and deontic operators does not result in modal collapse.

lemma $\lfloor P \rightarrow \mathbf{O}_a P \rfloor^D$ **nitpick oops**
lemma $\lfloor P \rightarrow \Box_a P \rfloor^D$ **nitpick oops**

Next we show that the necessitation rule does not work for indexical validity (in contrast to classical modal validity as defined for DDL).

lemma *NecLDa*: $\lfloor A \rfloor^D \Longrightarrow \lfloor \Box_a A \rfloor^D$ **nitpick oops**
lemma *NecLDp*: $\lfloor A \rfloor^D \Longrightarrow \lfloor \Box_p A \rfloor^D$ **nitpick oops**

The following can be seen as a kind of 'analytic/a priori necessity' operator (to be contrasted to the more traditional alethic necessity). In Kaplan's framework, a sentence being logically (i.e. indexically) valid means its being true *a priori*: it is guaranteed to be true in every possible context in which it is uttered, even though it may express distinct propositions in different contexts. This correlation between indexical validity and *a prioricity* has also been claimed in other two-dimensional semantic frameworks [10].

abbreviation *ldvalidbox* :: $m \Rightarrow m$ (‹$\Box^D$-› $[52]53$) **where** $\Box^D \varphi \equiv \lambda c\ w.\ \lfloor \varphi \rfloor^D$ — notice the D superscript
lemma $\lfloor \Box^D \varphi \rfloor_C \equiv \forall\, c. \lfloor \varphi \rfloor_c$ **by** *simp* — this operator works analogously to the box operator in modal logic S5

Quite trivially, the necessitation rule works for the combination of indexical validity with the previous operator.

lemma *NecLD*: $\lfloor A \rfloor^D \Longrightarrow \lfloor \Box^D A \rfloor^D$ **by** *simp*

The operator above is not part of the original Kaplan's LD ([7]) and has been added by us in order to better highlight some semantic features of our formalisation of Gewirth's argument in the next section and to being able to use the necessitation rule for some inference steps.

3.3 Quantification

We also enrich our logic with (higher-order) quantifiers (using parameterised types).

abbreviation *mforall*::$('t \Rightarrow m) \Rightarrow m$ (‹$\forall$›) **where** $\forall\, \Phi \equiv \lambda c\ w. \forall\, x.\ (\Phi\ x\ c\ w)$
abbreviation *mexists*::$('t \Rightarrow m) \Rightarrow m$ (‹$\exists$›) **where** $\exists\, \Phi \equiv \lambda c\ w. \exists\, x.\ (\Phi\ x\ c\ w)$
abbreviation *mforallBinder*::$('t \Rightarrow m) \Rightarrow m$ (**binder**‹$\forall$›$[8]9$) **where** $\forall\, x.\ (\varphi\ x) \equiv \forall\, \varphi$
abbreviation *mexistsBinder*::$('t \Rightarrow m) \Rightarrow m$ (**binder**‹$\exists$›$[8]9$) **where** $\exists\, x.\ (\varphi\ x) \equiv \exists\, \varphi$

Before starting our formalisation in the next section. We show that the axioms defined so far are consistent. Rather surprisingly, the *nunchaku* model finder states that no model has been found, while *nitpick* is indeed able to find one:

lemma *True* **nunchaku**[*satisfy*] **nitpick**[*satisfy*] **oops**

4 Gewith's Argument for the Principle of Generic Consistency (PGC)

Alan Gewirth's meta-ethical position is known as moral (or ethical) rationalism. According to it, moral principles are knowable *a priori*, by reason alone. Immanuel Kant is perhaps the most famous figure who has defended such a position. He has argued for the existence of upper moral principles (e.g. his "categorical imperative") from which we can reason (in a top-down fashion) in order to deduce and evaluate other more concrete maxims and actions. In contrast to Kant, Gewirth attempts to derive such upper moral principles by starting from non-moral considerations alone, namely from an agent's self-reflection. Gewirth's Principle of Generic Consistency (PGC) asserts that any agent (by virtue of its self-understanding as an agent) is rationally committed to asserting that (i) it has rights to freedom and well-being, and (ii) that all other agents have those same rights. Gewirth claims that, in his informal proof, the latter generalisation step (from "I" to all individuals) is done on purely logical grounds and does not presuppose any kind of universal moral principle. Gewirth's result is thus meant to hold with some kind of apodicticity (i.e. necessity). Deryck Beyleveld, author of an authoritative book on Gewirth's argument, puts it this way: "The argument purports to establish the PGC as a rationally necessary proposition with an apodictic status *for any PPA* equivalent to that enjoyed by the logical principle of noncontradiction itself." ([4] p. 1) If this is correct, then he succeeded in the task that Kant set himself, i.e. to found certain basic principles of morality in reason alone.

The argument for the PGC employs what Gewirth calls "the dialectically necessary method" within the "internal viewpoint" (perspective) of an agent. Although the drawn inferences are relative to the reasoning agent, Gewirth further argues that "the dialectically necessary method propounds the contents of this relativity as necessary ones, since the statements it presents reflect judgements all agents necessarily make on the basis of what is necessarily involved in their actions ... The statements the method attributes to the agent are set forth as necessary ones in that they reflect what is conceptually necessary to being an agent who voluntarily or freely acts for purposes he wants to attain." ([6]). In other words, the "dialectical necessity" of the assertions and inferences made in the argument comes from the definitional features (conceptual analysis) of the involved notions of agency, purposeful action, obligation, rights, etc. Hence the alternative notions of logical (i.e. indexical) validity and 'a priori necessity', developed in Kaplan's logical framework LD, have been considered by us as appropriate to model this kind of "dialectical necessity".

4.1 Conceptual Explications

type-synonym $p = e \Rightarrow m$ — Type for properties (function from individuals to sentence meanings)

4.1.1 Agency

The type chosen to represent what Gewirth calls "purposes" is not essential for the argument's validity. We choose to give "purposes" the same type as sentence meanings (type 'm'), so "acting on a purpose" would be represented in an analogous way to having a certain propositional attitude (e.g. "desiring that some proposition obtains").

consts *ActsOnPurpose*:: $e \Rightarrow m \Rightarrow m$ — ActsOnPurpose(A,E) gives the meaning of the sentence "A is

acting on purpose E"
consts *NeedsForPurpose*:: $e \Rightarrow p \Rightarrow m \Rightarrow m$ — NeedsForPurpose(A,P,E) gives the meaning of "A needs to have property P in order to reach purpose E"

In Gewirth's argument, an individual with agency (i.e. capable of purposive action) is said to be a PPA (prospective purposive agent).

definition *PPA*:: *p* **where** $PPA\ a \equiv \exists\, E.\ ActsOnPurpose\ a\ E$ — Definition of PPA

We have added the following axiom in order to guarantee the argument's logical correctness. It basically says that being a PPA is identity-constitutive for an individual (i.e. it's an essential property).

axiomatization where *essentialPPA*: $\lfloor \forall\, a.\ PPA\ a \rightarrow \Box^D(PPA\ a)\rfloor^D$ — being a PPA is an essential property

Quite interestingly, the axiom above entails, as a corollary, a kind of ability for a PPA to recognise other PPAs. For instance, if some individual holds itself as a PPA (i.e. seen from its own perspective/context 'd') then this individual (Agent(d)) is considered as a PPA from any other agent's perspective/context 'c'.

lemma *recognizeOtherPPA*: $\forall\, c\ d.\ \lfloor PPA\ (Agent\ d)\rfloor_d \longrightarrow \lfloor PPA\ (Agent\ d)\rfloor_c$ **using** *essentialPPA* **by** *blast*

4.1.2 Goodness

Gewirth's concept of (subjective) goodness, as employed in his argument, applies to purposes and is relative to some agent. It is therefore modelled as a binary relation relating an individual (type 'e') with a purpose (type 'm'). Other readings given by Gewirth's for the expression "P is good for A" include among others: "A attaches a positive value to P", "A values P proactively" and "A is motivated to achieve P".

consts *Good*::$e \Rightarrow m \Rightarrow m$

The following axioms interrelate the concept of goodness with the concept of agency, thus providing the above concepts with some meaning (by framing their inferential roles). Notice that such meaning-constitutive axioms (which we call "explications") are given as indexically valid (i.e. a priori) sentences.

axiomatization where *explicationGoodness1*: $\lfloor \forall\, a\ P.\ ActsOnPurpose\ a\ P \rightarrow Good\ a\ P\rfloor^D$
axiomatization where *explicationGoodness2*: $\lfloor \forall\, P\ M\ a.\ Good\ a\ P \wedge NeedsForPurpose\ a\ M\ P \rightarrow Good\ a\ (M\ a)\rfloor^D$
axiomatization where *explicationGoodness3*: $\lfloor \forall\, \varphi\ a.\ \Diamond_p \varphi \rightarrow \mathbf{O}\langle \varphi \mid \Box^D Good\ a\ \varphi\rangle\rfloor^D$

Below we show that all axioms defined so far are consistent:

lemma *True* **nitpick**[*satisfy*, *card c = 1*, *card e = 1*, *card w = 1*] **oops** — one-world model found (card w=1)

The first two assertions above have been explicitly provided by Gewirth as premises of his argument. The third axiom, however, has been added by us as an implicit premise in order to render Gewirth's proof as correct. This axiom aims at representing the intuitive notion of "seeking the good". In particular, it asserts that, from the point of view of an agent, necessarily good purposes are not only action motivating, but also entail an instrumental obligation to their realisation. The notion of necessity here involved is not the usual alethic one (which is represented in DDL with the modal box operators $\Box_a$ and $\Box_p$), but the linguistic one introduced above ($\Box^D$) derived from indexical validity, signaling that an agent holds some purpose as being true almost 'by definition' (i.e. a priori). This sets quite high standards for the kind of purposes an agent would ever take to be (instrumentally) obligatory and is indeed the weakest implicit premise we could come up with so far (taking away the $\Box^D$ 'a priori necessity' operator would indeed make this premise much stronger and our proof less credible).

4.1.3 Freedom and Well-Being

According to Gewirth, enjoying freedom and well-being (which we take together as a predicate: FWB) is the property which represents the "necessary conditions" or "generic features" of agency (i.e. being capable of purposeful action). Gewirth argues, the property of enjoying freedom and well-being (FWB) is special amongst other action-enabling properties, in that it is always required in order to act on any purpose (no matter which one).

consts *FWB*::*p* — Enjoying freedom and well-being (FWB) is a property (i.e. has type $e \Rightarrow m$)

axiomatization where
explicationFWB1: $\lfloor \forall\, P\ a.\ \mathit{NeedsForPurpose}\ a\ \mathit{FWB}\ P \rfloor^D$

We use model finder *nitpick* to verify that all axioms defined so far are consistent. *Nitpick* can indeed find a 'small' model with cardinality one for the sets of worlds and contexts.

lemma *True* **nitpick**[*satisfy*, *card c = 1*, *card e = 1*, *card w = 1*] **oops** — one-world model found

At some point in Gewirth's argument we have to show that there exists an (instrumental) obligation to enjoying freedom and well-being (FWB). Since, according to the so-called "Kant's law" (which is a corollary of DDL), impossible or necessary things cannot be obligatory, we can reasonably demand that FWB be (metaphysically) possible for every agent. As before, we take this demand to be an a priori characteristic of the concept of FWB and therefore axiomatise it as an indexically valid sentence.

axiomatization where *explicationFWB2*: $\lfloor \forall\, a.\ \Diamond_p\ \mathit{FWB}\ a \rfloor^D$
axiomatization where *explicationFWB3*: $\lfloor \forall\, a.\ \Diamond_p\ \neg \mathit{FWB}\ a \rfloor^D$

As a result of enforcing the contingency of FWB, the models found by *nitpick* now have a cardinality of two for the set of worlds:

lemma *True* **nitpick**[*satisfy*, *card c = 1*, *card e = 1*, *card w = 1*, *expect=none*] **oops** — no model found for one-world models
lemma *True* **nitpick**[*satisfy*, *card c = 1*, *card e = 1*, *card w = 2*] **oops** — models need now at least two worlds

4.1.4 Obligation and Interference

Kant's Law ("ought implies can") is derivable directly from DDL: If φ oughts to obtain then φ is possible. Note that we will use for the formalisation of Gewirth's argument the DDL ideal obligation operator ($\mathbf{O}_i$) but we could have also used (mutatis mutandis) the DDL actual obligation operator ($\mathbf{O}_a$).

lemma $\lfloor \mathbf{O}_i\varphi \rightarrow \Diamond_p\varphi \rfloor$ **using** *sem-5ab* **by** *simp*

Furthermore, we have seen the need to postulate the following (implicit) premise in order to validate the argument. This axiom can be seen as a variation of the so-called Kant's law ("ought implies can"), i.e. an impossible act cannot be obligatory. In the same vein, our variation can be read as "ought implies ought to can" and is closer to Gewirth's own description: that having an obligation to do X implies that "I ought (in the same sense and the same criterion) to be free to do X, that I ought not to be prevented from doing X, that my capacity to do X ought not to be interfered with." ([6] p. 91-95)

axiomatization where *OIOAC*: $\lfloor \mathbf{O}_i\varphi \rightarrow \mathbf{O}_i(\Diamond_a\varphi)\rfloor^D$

Concerning the concept of interference, we state that the existence of an individual (successfully) interfering with some state of affairs S implies that S cannot possibly obtain in any of the actually possible situations (and the other way round). Note that for this definition we have employed a possibility operator ($\Diamond_a$) which is weaker than metaphysical possibility ($\Diamond_p$) (see Carmo and Jones DDL framework [5] for details). Also note that we have also employed the (stronger) classical notion of modal validity instead of indexical validity. (So far we haven't been able to get theorem provers and model finders to prove/disprove Gewirth's proof if formalizing this axiom as simply indexically valid.)

consts *InterferesWith*::$e\Rightarrow m\Rightarrow m$ — an individual can interfere with some state of affairs (from obtaining)
axiomatization where *explicationInterference*: $\lfloor(\exists\, b.\ \mathit{InterferesWith}\ b\ \varphi) \leftrightarrow \neg\Diamond_a\varphi\rfloor$

From the previous axiom we can prove following corollaries: If someone (successfully) interferes with agent 'a' having FWB, then 'a' can no longer possibly enjoy its FWB (and the other way round).

lemma $\lfloor\forall\, a.\ (\exists\, b.\ \mathit{InterferesWith}\ b\ (FWB\ a)) \leftrightarrow \neg\Diamond_a(FWB\ a)\rfloor$ **using** *explicationInterference* **by** *blast*
lemma *InterferenceWithFWB*: $\lfloor\forall\, a.\ \ \Diamond_a(FWB\ a) \leftrightarrow (\forall\, b.\ \neg\mathit{InterferesWith}\ b\ (FWB\ a))\rfloor$ **using** *explicationInterference* **by** *blast*

4.1.5 Rights and Other-Directed Obligations

Gewirth points out the existence of a correlation between an agent's own claim rights and other-referring obligations (see e.g. [6], p. 66). A claim right is a right which entails duties or obligations on other agents regarding the right-holder (so-called Hohfeldian claim rights in legal theory). We model this concept of claim rights in such a way that an individual 'a' has a (claim) right to some property 'P' if and only if it is obligatory that every (other) individual 'b' does not interfere with the state of affairs 'P(a)' from obtaining. Since there is no particular individual to whom this directive is addressed, this obligation has been referred to by Gewirth as being "other-directed" (aka. "other-referring") in contrast to "other-directing" obligations which entail a moral obligation for some particular subject ([4] p. 41,51). This latter distinction is essential to Gewirth's argument.

definition *RightTo*::$e\Rightarrow(e\Rightarrow m)\Rightarrow m$ **where** $\mathit{RightTo}\ a\ \varphi \equiv \mathbf{O}_i(\forall\, b.\ \neg\mathit{InterferesWith}\ b\ (\varphi\ a))$

Now that all needed axioms and definitions are in place, we use model finder *nitpick* to show that they are consistent:

lemma *True* **nitpick**[*satisfy*, *card c = 1*, *card e = 1*, *card w = 2*] **oops** — models with at least two worlds found

4.2 Formal Proof of Gewirth's Argument for the PGC

Following Beyleveld's summary ([4], ch. 2), the main steps of the argument are (with original numbering):

(1) I act voluntarily for some (freely chosen) purpose E (equivalent –by definition– to: I am a PPA).

(2) E is (subjectively) good (i.e. I value E proactively).

(3) My freedom and well-being (FWB) are generically necessary conditions of my agency (i.e. I need them to achieve any purpose whatsoever).

(4) My FWB are necessary goods (at least for me).

(5) I (even if no one else) have a claim right to my FWB.

(13) Every PPA has a claim right to their FWB.

4.2.1 Weak Variant

In the following we present a formalised proof for a weak variant of the Principle of Generic Consistency (PGC), which asserts that the following sentence is valid from every PPA's standpoint: "I (as a PPA) have a claim right to my freedom and well-being".

theorem *PGC-weak*: **shows** $\forall\, C.\ \lfloor PPA\ (Agent\ C) \rightarrow (RightTo\ (Agent\ C)\ FWB)\rfloor_C$
proof – **{**
fix $C{::}c$ — 'C' is some arbitrarily chosen context (agent's perspective)
let $?I = (Agent\ C)$ — 'I' is/am the agent with perspective 'C'
{
fix $E{::}m$ — 'E' is some arbitrarily chosen purpose
{
assume *P1*: $\lfloor ActsOnPurpose\ ?I\ E\rfloor_C$ — (1) I act voluntarily on purpose E
from *P1* **have** *P1a*: $\lfloor PPA\ ?I\rfloor_C$ **using** *PPA-def* **by** *auto* — (1a) I am a PPA
from *P1* **have** *C2*: $\lfloor Good\ ?I\ E\rfloor_C$ **using** *explicationGoodness1 essentialPPA* **by** *meson* — (2) purpose E is good for me
from *explicationFWB1* **have** *C3*: $\lfloor\forall\, P.\ NeedsForPurpose\ ?I\ FWB\ P\rfloor^D$ **by** *simp* — (3) I need FWB for any purpose whatsoever
hence $\exists\, P.\lfloor Good\ ?I\ P \land NeedsForPurpose\ ?I\ FWB\ P\rfloor^D$ **using** *explicationFWB2 explicationGoodness3 sem-5ab* **by** *blast*
hence $\lfloor Good\ ?I\ (FWB\ ?I)\rfloor^D$ **using** *explicationGoodness2* **by** *blast* — FWB is (a priori) good for me (in a kind of definitional sense)
hence *C4*: $\lfloor\Box^D(Good\ ?I\ (FWB\ ?I))\rfloor_C$ **by** *simp* — (4) FWB is an (a priori) necessary good for me
have $\lfloor\mathbf{O}\langle FWB\ ?I \mid \Box^D(Good\ ?I)\ (FWB\ ?I)\rangle\rfloor_C$ **using** *explicationGoodness3 explicationFWB2* **by** *blast* — I ought to pursue my FWB on the condition that I consider it to be a necessary good
hence $\lfloor\mathbf{O}_i(FWB\ ?I)\rfloor_C$ **using** *explicationFWB2 explicationFWB3 C4 CJ-14p* **by** *fastforce* — There is an (other-directed) obligation to my FWB
hence $\lfloor\mathbf{O}_i(\Diamond_a(FWB\ ?I))\rfloor_C$ **using** *OIOAC* **by** *simp* — It must therefore be the case that my FWB is possible
hence $\lfloor\mathbf{O}_i(\forall\, a.\ \neg InterferesWith\ a\ (FWB\ ?I))\rfloor_C$ **using** *InterferenceWithFWB* **by** *simp* — There is an obligation for others not to interfere with my FWB
hence *C5*: $\lfloor RightTo\ ?I\ FWB\rfloor_C$ **using** *RightTo-def* **by** *simp* — (5) I have a claim right to my freedom and well-being

}
hence $\lfloor$*ActsOnPurpose ?I E* $\rightarrow$ *RightTo ?I FWB*$\rfloor_C$ **by** (*rule impI*) — I have a claim right to my freedom and well-being (since I act on some purpose E)
}
hence $\lfloor\forall$ *P. ActsOnPurpose ?I P* $\rightarrow$ *RightTo ?I FWB*$\rfloor_C$ **by** (*rule allI*) — "allI" is a logical generalisation rule: "all-quantifier introduction"
hence $\lfloor$*PPA ?I* $\rightarrow$ *RightTo ?I FWB*$\rfloor_C$ **using** *PPA-def* **by** *simp* — (seen from my perspective C) I have a claim right to my freedom and well-being since I am a PPA
hence $\lfloor$*PPA* (*Agent C*) $\rightarrow$ *RightTo* (*Agent C*) *FWB*$\rfloor_C$ **by** *simp* — (seen from the perspective C) C's agent has a claim right to its freedom and well-being since it is a PPA
}
thus *C13*: $\forall$ *C*. $\lfloor$*PPA* (*Agent C*) $\rightarrow$ (*RightTo* (*Agent C*) *FWB*)$\rfloor_C$ **by** (*rule allI*) — (13) For every perspective C: C's agent has a claim right to its freedom and well-being
qed

Regarding the last inference step, given that the context (agent's perspective) 'C' has been arbitrarily fixed at the beginning, we can use again the "all-quantifier introduction" rule to generalise the previous assertion to all possible contexts 'C' (and agents 'Agent(C)'). Note that the generalisation from "I" to all individuals has been done on purely logical grounds and does not involve any kind of universal moral principle. This is a main requirement Gewirth has set for his argument.

4.2.2 Strong Variant

This is a proof for a stronger variant of the PGC, which asserts that the following sentence is valid from every PPA's standpoint: "Every PPA has a claim right to its freedom and well-being (FWB)".

theorem *PGC-strong*: **shows** $\lfloor\forall$ *x*. *PPA x* $\rightarrow$ (*RightTo x FWB*)$\rfloor^D$
proof – {
fix *C*::*c* — 'C' is some arbitrarily chosen context (agent's perspective)
{
fix *I*::*e* — 'I' is some arbitrarily chosen individual (agent's perspective)
{
fix *E*::*m* — 'E' is some arbitrarily chosen purpose
{
assume *P1*: $\lfloor$*ActsOnPurpose I E*$\rfloor_C$ — (1) I act voluntarily on purpose E
from *P1* **have** *P1a*: $\lfloor$*PPA I*$\rfloor_C$ **using** *PPA-def* **by** *auto* — (1a) I am a PPA
from *P1* **have** *C2*: $\lfloor$*Good I E*$\rfloor_C$ **using** *explicationGoodness1 essentialPPA* **by** *meson* — (2) purpose E is good for me
from *explicationFWB1* **have** *C3*: $\lfloor\forall$ *P*. *NeedsForPurpose I FWB P*$\rfloor^D$ **by** *simp* — (3) I need FWB for any purpose whatsoever
hence $\exists$ *P*.$\lfloor$*Good I P* $\wedge$ *NeedsForPurpose I FWB P*$\rfloor^D$ **using** *explicationFWB2 explicationGoodness3 sem-5ab* **by** *blast*
hence $\lfloor$*Good I* (*FWB I*)$\rfloor^D$ **using** *explicationGoodness2* **by** *blast* — FWB is (a priori) good for me (in a kind of definitional sense)
hence *C4*: $\lfloor\Box^D$(*Good I* (*FWB I*))$\rfloor_C$ **by** *simp* — (4) FWB is an (a priori) necessary good for me
have $\lfloor\mathbf{O}\langle$*FWB I* $\mid$ $\Box^D$(*Good I*) (*FWB I*)$\rangle\rfloor_C$ **using** *explicationGoodness3 explicationFWB2* **by** *blast* — I ought to pursue my FWB on the condition that I consider it to be a necessary good
hence $\lfloor\mathbf{O}_i$(*FWB I*)$\rfloor_C$ **using** *explicationFWB2 explicationFWB3 C4 CJ-14p* **by** *fastforce* — There is an (other-directed) obligation to my FWB

hence $\lfloor \mathbf{O}_i(\Diamond_a(FWB\ I)) \rfloor_C$ **using** *OIOAC* **by** *simp* — It must therefore be the case that my FWB is possible
hence $\lfloor \mathbf{O}_i(\forall a.\ \neg InterferesWith\ a\ (FWB\ I)) \rfloor_C$ **using** *InterferenceWithFWB* **by** *simp* — There is an obligation for others not to interfere with my FWB
hence *C5*: $\lfloor RightTo\ I\ FWB \rfloor_C$ **using** *RightTo-def* **by** *simp* — (5) I have a claim right to my FWB
}
hence $\lfloor ActsOnPurpose\ I\ E \rightarrow RightTo\ I\ FWB \rfloor_C$ **by** (*rule impI*) — I have a claim right to my FWB (since I act on some purpose E)
}
hence $\lfloor \forall P.\ ActsOnPurpose\ I\ P \rightarrow RightTo\ I\ FWB \rfloor_C$ **by** (*rule allI*)
hence $\lfloor PPA\ I \rightarrow RightTo\ I\ FWB \rfloor_C$ **using** *PPA-def* **by** *simp* — I have a claim right to my FWB since I am a PPA
}
hence $\forall x.\ \lfloor PPA\ x \rightarrow RightTo\ x\ FWB \rfloor_C$ **by** *simp* — Every agent has a claim right to its FWB since it is a PPA
}
thus *C13*: $\forall C.\ \lfloor \forall x.\ PPA\ x \rightarrow (RightTo\ x\ FWB) \rfloor_C$ **by** (*rule allI*) — (13) For every perspective C: every agent has a claim right to its FWB
qed

We show that the weaker variant of the PGC presented above can be derived from the stronger one.

lemma *PGC-weak2*: $\forall C.\ \lfloor PPA\ (Agent\ C) \rightarrow (RightTo\ (Agent\ C)\ FWB) \rfloor_C$ **using** *PGC-strong* **by** *simp*

4.2.3 Some Exemplary Inferences

In the following, we illustrate how to draw some inferences building upon Gewirth's PGC.

consts $X{::}c$ — Context of use X (to which a certain speaker agent corresponds)
consts $Y{::}c$ — Context of use Y (to which another speaker agent corresponds)

The agent (of context) X holds itself as a PPA.

axiomatization where *AgentX-X-PPA*: $\lfloor PPA\ (Agent\ X) \rfloor_X$

The agent (of another context) Y holds the agent (of context) X as a PPA.

lemma *AgentY-X-PPA*: $\lfloor PPA\ (Agent\ X) \rfloor_Y$ **using** *AgentX-X-PPA recognizeOtherPPA* **by** *simp*

Now the agent (of context) Y holds itself as a PPA.

axiomatization where *AgentY-Y-PPA*: $\lfloor PPA\ (Agent\ Y) \rfloor_Y$

The agent Y claims a right to FWB.

lemma *AgentY-Y-FWB*: $\lfloor RightTo\ (Agent\ Y)\ FWB \rfloor_Y$ **using** *AgentY-Y-PPA PGC-weak* **by** *simp*

The agent Y accepts X claiming a right to FWB.

lemma *AgentY-X-FWB*: $\lfloor RightTo\ (Agent\ X)\ FWB \rfloor_Y$ **using** *AgentY-X-PPA PGC-strong* **by** *simp*

The agent Y accepts an (other-directed) obligation of non-interference with X's FWB.

lemma *AgentY-NonInterference-X-FWB*: $\lfloor \mathbf{O}_i(\forall z.\ \neg InterferesWith\ z\ (FWB\ (Agent\ X))) \rfloor_Y$ **using** *AgentY-X-FWB RightTo-def* **by** *simp*

Axiom consistency checked: Nitpick finds a two-world model (card w=2).

lemma *True* **nitpick**[*satisfy*, *card c = 1*, *card e = 1*, *card w = 2*] **oops**

References

[1] C. Benzmüller, A. Farjami, and X. Parent. A dyadic deontic logic in HOL. In J. Broersen, C. Condoravdi, S. Nair, and G. Pigozzi, editors, *Deontic Logic and Normative Systems — 14th International Conference, DEON 2018, Utrecht, The Netherlands, 3-6 July, 2018*, pages 33–50. College Publications, 2018.

[2] C. Benzmüller, A. Farjami, and X. Parent. Faithful semantical embedding of a dyadic deontic logic in HOL. Technical report, CoRR, 2018. https://arxiv.org/abs/1802.08454.

[3] C. Benzmüller and L. Paulson. Quantified multimodal logics in simple type theory. *Logica Universalis (Special Issue on Multimodal Logics)*, 7(1):7–20, 2013.

[4] D. Beyleveld. *The dialectical necessity of morality: An analysis and defense of Alan Gewirth's argument to the principle of generic consistency.* University of Chicago Press, 1991.

[5] J. Carmo and A. J. Jones. Deontic logic and contrary-to-duties. In *Handbook of philosophical logic*, pages 265–343. Springer, 2002.

[6] A. Gewirth. *Reason and morality.* University of Chicago Press, 1981.

[7] D. Kaplan. On the logic of demonstratives. *Journal of philosophical logic*, 8(1):81–98, 1979.

[8] D. Kaplan. Afterthoughts. 1989.

[9] A. Kornai. Bounding the impact of AGI. *Journal of Experimental & Theoretical Artificial Intelligence*, 26(3):417–438, 2014.

[10] L. Schroeter. Two-dimensional semantics. In E. N. Zalta, editor, *The Stanford Encyclopedia of Philosophy.* Metaphysics Research Lab, Stanford University, summer 2017 edition, 2017.