Topological semantics for paraconsistent and paracomplete logics

David Fuenmayor

December 14, 2021

Abstract

We introduce a generalized topological semantics for paraconsistent and paracomplete logics by drawing upon early works on topological Boolean algebras (cf. works by Kuratowski, Zarycki, McKinsey & Tarski, etc.). In particular, this work exemplarily illustrates the shallow semantical embeddings approach (SSE) employing the proof assistant Isabelle/HOL. By means of the SSE technique we can effectively harness theorem provers, model finders and 'hammers' for reasoning with quantified non-classical logics.

Contents

theory *sse-boolean-algebra*
imports *Main*
begin

declare[[*syntax-ambiguity-warning=false*]]
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

1 Shallow embedding of a Boolean algebra of propositions

In this section we present a shallow semantical embedding (SSE, cf. [1] and [2]) for a family of logics whose semantics is based upon extensions of (complete and atomic) Boolean algebras. The range of such logics is indeed very wide, including, as we will see, quantified paraconsistent and paracomplete (e.g. intuitionistic) logics. Aside from illustrating how the SSE approach works in practice we show how it allows us to effectively harness theorem provers, model finders and 'hammers' for reasoning with quantified non-classical logics. Proof reconstructions have deliberately been avoided. Most of the proofs (in fact, all one-liners) have been found using Sledgehammer.

Two notions play a fundamental role in this work: propositions and propositional functions. Propositions, qua sentence denotations, are modeled as objects of type $w \Rightarrow bool$ (shortened as σ). Propositional functions, as the name indicates, are basically anything with a (parametric) type $'t \Rightarrow \sigma$.

We introduce a type w for the domain of points (aka. 'worlds', 'states', etc.). σ is a type alias for sets of points (i.e. propositions) encoded as characteristic functions.

typedecl *w*
type-synonym $\sigma = w \Rightarrow bool$

In the sequel, we introduce the following naming convention for variables:

(i) Latin letters (A, b, M, P, q, X, y, etc.) denote in general propositions (type σ); however, we reserve letters D and S to denote sets of propositions (aka. domains/spaces) and the letters u, v and w to denote worlds/points.

(ii) Greek letters (in particular π) denote propositional functions (type $'t \Rightarrow \sigma$); among the latter we may employ the letters φ, ψ and η to explicitly name those corresponding to unary connectives/operations (type $\sigma \Rightarrow \sigma$).

1.1 Encoding Boolean operations

We start with an ordered algebra,

abbreviation *sequ*::$\sigma \Rightarrow \sigma \Rightarrow bool$ (**infixr** $\approx$ *45*) **where** $A \approx B \equiv \forall w.\ (A\ w) \longleftrightarrow (B\ w)$
abbreviation *subs*::$\sigma \Rightarrow \sigma \Rightarrow bool$ (**infixr** $\preceq$ *45*) **where** $A \preceq B \equiv \forall w.\ (A\ w) \longrightarrow (B\ w)$
abbreviation *sups*::$\sigma \Rightarrow \sigma \Rightarrow bool$ (**infixr** $\succeq$ *45*) **where** $B \succeq A \equiv A \preceq B$

define meet and join by reusing HOL metalogical conjunction and disjunction,

definition *meet*::$\sigma \Rightarrow \sigma \Rightarrow \sigma$ (**infixr** $\wedge$ *54*) **where** $A \wedge B \equiv \lambda w.\ (A\ w) \wedge (B\ w)$
definition *join*::$\sigma \Rightarrow \sigma \Rightarrow \sigma$ (**infixr** $\vee$ *53*) **where** $A \vee B \equiv \lambda w.\ (A\ w) \vee (B\ w)$

and introduce further operations to obtain a Boolean 'algebra of propositions'.

definition *top*::σ ($\top$) **where** $\top \equiv \lambda w.\ True$
definition *bottom*::σ ($\bot$) **where** $\bot \equiv \lambda w.\ False$
definition *impl*::$\sigma \Rightarrow \sigma \Rightarrow \sigma$ (**infixr** $\rightarrow$ *51*) **where** $A \rightarrow B \equiv \lambda w.\ (A\ w) \longrightarrow (B\ w)$
definition *dimp*::$\sigma \Rightarrow \sigma \Rightarrow \sigma$ (**infixr** $\leftrightarrow$ *51*) **where** $A \leftrightarrow B \equiv \lambda w.\ (A\ w) \longleftrightarrow (B\ w)$
definition *diff*::$\sigma \Rightarrow \sigma \Rightarrow \sigma$ (**infixr** $\leftharpoonup$ *51*) **where** $A \leftharpoonup B \equiv \lambda w.\ (A\ w) \wedge \neg(B\ w)$
definition *compl*::$\sigma \Rightarrow \sigma$ ($-$- [*57*]*58*) **where** $-A \equiv \lambda w.\ \neg(A\ w)$

named-theorems *conn*
declare *meet-def*[*conn*] *join-def*[*conn*] *top-def*[*conn*] *bottom-def*[*conn*]
impl-def[*conn*] *dimp-def*[*conn*] *diff-def*[*conn*] *compl-def*[*conn*]

Quite trivially, we can verify that the algebra satisfies some essential lattice properties.

lemma $a \vee a \approx a$ ⟨*proof*⟩
lemma $a \wedge a \approx a$ ⟨*proof*⟩
lemma $a \preceq a \vee b$ ⟨*proof*⟩
lemma $a \wedge b \preceq a$ ⟨*proof*⟩
lemma $(a \wedge b) \vee b \approx b$ ⟨*proof*⟩
lemma $a \wedge (a \vee b) \approx a$ ⟨*proof*⟩
lemma $a \preceq c \Longrightarrow b \preceq c \Longrightarrow a \vee b \preceq c$ ⟨*proof*⟩
lemma $c \preceq a \Longrightarrow c \preceq b \Longrightarrow c \preceq a \wedge b$ ⟨*proof*⟩
lemma $a \preceq b \equiv (a \vee b) \approx b$ ⟨*proof*⟩
lemma $b \preceq a \equiv (a \wedge b) \approx b$ ⟨*proof*⟩
lemma $a \preceq c \Longrightarrow b \preceq d \Longrightarrow (a \vee b) \preceq (c \vee d)$ ⟨*proof*⟩
lemma $a \preceq c \Longrightarrow b \preceq d \Longrightarrow (a \wedge b) \preceq (c \wedge d)$ ⟨*proof*⟩

1.2 Second-order operations and fixed-points

We define equality for propositional functions as follows.

definition *equal-op*::$('t \Rightarrow \sigma) \Rightarrow ('t \Rightarrow \sigma) \Rightarrow bool$ (**infix** $\equiv$ *60*) **where** $\varphi \equiv \psi \equiv \forall X.\ \varphi\ X \approx \psi\ X$

Moreover, we define some useful Boolean (2nd-order) operations on propositional functions,

abbreviation *unionOp*::$('t \Rightarrow \sigma) \Rightarrow ('t \Rightarrow \sigma) \Rightarrow ('t \Rightarrow \sigma)$ (**infixr** $\sqcup$ *61*) **where** $\varphi \sqcup \psi \equiv \lambda X.\ \varphi\ X \vee \psi\ X$
abbreviation *interOp*::$('t \Rightarrow \sigma) \Rightarrow ('t \Rightarrow \sigma) \Rightarrow ('t \Rightarrow \sigma)$ (**infixr** $\sqcap$ *62*) **where** $\varphi \sqcap \psi \equiv \lambda X.\ \varphi\ X \wedge \psi\ X$
abbreviation *compOp*::$('t \Rightarrow \sigma) \Rightarrow ('t \Rightarrow \sigma)$ ((-^{c})) **where** $\varphi^c \equiv \lambda X.\ -\varphi\ X$

some of them explicitly targeting operations,

definition *dual*::$(\sigma \Rightarrow \sigma) \Rightarrow (\sigma \Rightarrow \sigma)$ ((-^{d})) **where** $\varphi^d \equiv \lambda X.\ -(\varphi(-X))$

and also define an useful operation (for technical purposes).

definition *id*::$\sigma \Rightarrow \sigma$ (*id*) **where** $id\ A \equiv A$

We now prove some useful lemmas (some of them may help the provers in their hard work).

lemma *comp-symm*: $\varphi^c = \psi \Longrightarrow \varphi = \psi^c$ ⟨*proof*⟩
lemma *comp-invol*: $\varphi^{cc} = \varphi$ ⟨*proof*⟩
lemma *dual-symm*: $(\varphi \equiv \psi^d) \Longrightarrow (\psi \equiv \varphi^d)$ ⟨*proof*⟩
lemma *dual-comp*: $\varphi^{dc} = \varphi^{cd}$ ⟨*proof*⟩

lemma $id^d \equiv id$ ⟨*proof*⟩
lemma $id^c \equiv compl$ ⟨*proof*⟩
lemma $(A \sqcup B)^d \equiv (A^d) \sqcap (B^d)$ ⟨*proof*⟩
lemma $(A \sqcup B)^c \equiv (A^c) \sqcap (B^c)$ ⟨*proof*⟩
lemma $(A \sqcap B)^d \equiv (A^d) \sqcup (B^d)$ ⟨*proof*⟩
lemma $(A \sqcap B)^c \equiv (A^c) \sqcup (B^c)$ ⟨*proof*⟩

The notion of a fixed point is a fundamental one. We speak of propositions being fixed points of operations. For a given operation we define in the usual way a fixed-point predicate for propositions.

abbreviation *fixedpoint*::$(\sigma \Rightarrow \sigma) \Rightarrow (\sigma \Rightarrow bool)$ (fp) **where** $fp\ \varphi \equiv \lambda X.\ \varphi\ X \approx X$

lemma *fp-d*: $(fp\ \varphi^d)\ X = (fp\ \varphi)(-X)$ ⟨*proof*⟩
lemma *fp-c*: $(fp\ \varphi^c)\ X = (X \approx -(\varphi\ X))$ ⟨*proof*⟩
lemma *fp-dc*:$(fp\ \varphi^{dc})\ X = (X \approx \varphi(-X))$ ⟨*proof*⟩

Indeed, we can 'operationalize' this predicate by defining a fixed-point operator as follows:

abbreviation *fixedpoint-op*::$(\sigma \Rightarrow \sigma) \Rightarrow (\sigma \Rightarrow \sigma)$ $((\text{-}^{fp}))$ **where** $\varphi^{fp} \equiv \lambda X.\ (\varphi\ X) \leftrightarrow X$

lemma *ofp-c*: $(\varphi^c)^{fp} \equiv (\varphi^{fp})^c$ ⟨*proof*⟩
lemma *ofp-d*: $(\varphi^d)^{fp} \equiv (\varphi^{fp})^{dc}$ ⟨*proof*⟩
lemma *ofp-dc*:$(\varphi^{dc})^{fp} \equiv (\varphi^{fp})^d$ ⟨*proof*⟩
lemma *ofp-decomp*: $\varphi^{fp} \equiv (id \sqcap \varphi) \sqcup ((id \sqcup \varphi)^c)$ ⟨*proof*⟩
lemma *ofp-invol*: $(\varphi^{fp})^{fp} \equiv \varphi$ ⟨*proof*⟩

Fixed-point predicate and fixed-point operator are closely related.

lemma *fp-rel*: $((fp\ \varphi)\ X) = (\varphi^{fp}\ X \approx \top)$ ⟨*proof*⟩
lemma *fp-d-rel*: $((fp\ \varphi^d)\ X) = (\varphi^{fp}(-X) \approx \top)$ ⟨*proof*⟩
lemma *fp-c-rel*: $((fp\ \varphi^c)\ X) = (\varphi^{fp}\ X \approx \bot)$ ⟨*proof*⟩
lemma *fp-dc-rel*: $((fp\ \varphi^{dc})\ X) = (\varphi^{fp}(-X) \approx \bot)$ ⟨*proof*⟩

1.3 Equality and atomicity

We prove some facts about equality (which may help improve prover's performance).

lemma *eq-ext*: $a \approx b \Longrightarrow a = b$ ⟨*proof*⟩
lemma *eq-ext'*: $a \equiv b \Longrightarrow a = b$ ⟨*proof*⟩
lemma *meet-char*: $a \preceq b \longleftrightarrow a \wedge b \approx a$ ⟨*proof*⟩
lemma *join-char*: $a \preceq b \longleftrightarrow a \vee b \approx b$ ⟨*proof*⟩

We can verify indeed that the algebra is atomic (in three different ways) by relying on the presence of primitive equality in HOL. A more general class of Boolean algebras could in principle be obtained in systems without primitive equality or by suitably restricting quantification over propositions (e.g. defining a topology and restricting quantifiers to open/closed sets).

definition *atom* $a \equiv \neg(a \approx \bot) \wedge (\forall p.\ a \preceq p \vee a \preceq -p)$
lemma *atomic1*: $\forall w.\ \exists q.\ q\ w \wedge (\forall p.\ p\ w \longrightarrow q \preceq p)$ ⟨*proof*⟩
lemma *atomic2*: $\forall w.\ \exists q.\ q\ w \wedge atom(q)$ ⟨*proof*⟩
lemma *atomic3*: $\forall p.\ \neg(p \approx \bot) \longrightarrow (\exists q.\ atom(q) \wedge q \preceq p)$ ⟨*proof*⟩

end
theory *sse-boolean-algebra-quantification*
imports *sse-boolean-algebra*
begin
hide-const(**open**) *List.list.Nil* **no-notation** *List.list.Nil* ([])
hide-const(**open**) *Relation.converse* **no-notation** *Relation.converse* $((\text{-}^{-1})$ [*1000*] *999*)
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

1.4 Obtaining a complete Boolean Algebra

Our aim is to obtain a complete Boolean algebra which we can use to interpret quantified formulas (in the spirit of Boolean-valued models for set theory).

We start by defining infinite meet (infimum) and infinite join (supremum) operations,

definition *infimum*:: $(\sigma \Rightarrow bool) \Rightarrow \sigma$ ($\bigwedge$-) **where** $\bigwedge S \equiv \lambda w.\ \forall X.\ S\ X \longrightarrow X\ w$
definition *supremum*::$(\sigma \Rightarrow bool) \Rightarrow \sigma$ ($\bigvee$-) **where** $\bigvee S \equiv \lambda w.\ \exists X.\ S\ X\ \wedge\ X\ w$

and show that the corresponding lattice is complete.

abbreviation *upper-bound* $U\ S \equiv \forall X.\ (S\ X) \longrightarrow X \preceq U$
abbreviation *lower-bound* $L\ S \equiv \forall X.\ (S\ X) \longrightarrow L \preceq X$
abbreviation *is-supremum* $U\ S \equiv$ *upper-bound* $U\ S \wedge (\forall X.$ *upper-bound* $X\ S \longrightarrow U \preceq X)$
abbreviation *is-infimum* $L\ S \equiv$ *lower-bound* $L\ S \wedge (\forall X.$ *lower-bound* $X\ S \longrightarrow X \preceq L)$

lemma *sup-char*: *is-supremum* $\bigvee S\ S$ ⟨*proof*⟩
lemma *sup-ext*: $\forall S.\ \exists X.$ *is-supremum* $X\ S$ ⟨*proof*⟩
lemma *inf-char*: *is-infimum* $\bigwedge S\ S$ ⟨*proof*⟩
lemma *inf-ext*: $\forall S.\ \exists X.$ *is-infimum* $X\ S$ ⟨*proof*⟩

We can check that being closed under supremum/infimum entails being closed under join/meet.

abbreviation *meet-closed* $S \equiv\ \forall X\ Y.\ (S\ X \wedge S\ Y) \longrightarrow S(X \wedge\ Y)$
abbreviation *join-closed* $S \equiv\ \forall X\ Y.\ (S\ X \wedge S\ Y) \longrightarrow S(X \vee\ Y)$

abbreviation *nonEmpty* $S \equiv \exists x.\ S\ x$
abbreviation *contains* $S\ D \equiv\ \forall X.\ D\ X \longrightarrow S\ X$
abbreviation *infimum-closed* $S\ \equiv \forall D.$ *nonEmpty* $D \wedge$ *contains* $S\ D \longrightarrow S(\bigwedge D)$
abbreviation *supremum-closed* $S \equiv \forall D.$ *nonEmpty* $D \wedge$ *contains* $S\ D \longrightarrow S(\bigvee D)$

lemma *inf-meet-closed*: $\forall S.$ *infimum-closed* $S \longrightarrow$ *meet-closed* S ⟨*proof*⟩
lemma *sup-join-closed*: $\forall P.$ *supremum-closed* $P \longrightarrow$ *join-closed* P ⟨*proof*⟩

1.5 Adding quantifiers (restricted and unrestricted)

We can harness HOL to define quantification over individuals of arbitrary type (using polymorphism). These (unrestricted) quantifiers take a propositional function and give a proposition.

abbreviation *mforall*::$('t \Rightarrow \sigma) \Rightarrow \sigma$ ($\forall$ - [*55*]*56*) **where** $\forall \pi \equiv \lambda w.\ \forall X.\ (\pi\ X)\ w$
abbreviation *mexists*::$('t \Rightarrow \sigma) \Rightarrow \sigma$ ($\exists$ - [*55*]*56*) **where** $\exists \pi \equiv \lambda w.\ \exists X.\ (\pi\ X)\ w$

To improve readability, we introduce for them an useful binder notation.

abbreviation *mforallB* (**binder**$\forall$ [*55*]*56*) **where** $\forall X.\ \pi\ X \equiv \forall \pi$
abbreviation *mexistsB* (**binder**$\exists$ [*55*]*56*) **where** $\exists X.\ \pi\ X \equiv \exists \pi$

Moreover, we define restricted quantifiers which take a 'functional domain' as additional parameter. The latter is a propositional function that maps each element 'e' to the proposition 'e exists'.

abbreviation *mforall-restr*::$('t{\Rightarrow}\sigma){\Rightarrow}('t{\Rightarrow}\sigma){\Rightarrow}\sigma$ ($\forall^R$(-)-) **where** $\forall^R(\delta)\pi \equiv \lambda w.\forall X.\ (\delta\ X)\ w \longrightarrow (\pi\ X)\ w$
abbreviation *mexists-restr*::$('t{\Rightarrow}\sigma){\Rightarrow}('t{\Rightarrow}\sigma){\Rightarrow}\sigma$ ($\exists^R$(-)-) **where** $\exists^R(\delta)\pi \equiv \lambda w.\exists X.\ (\delta\ X)\ w\ \wedge\ (\pi\ X)\ w$

1.6 Relating quantifiers with further operators

The following 'type-lifting' function is useful for converting sets into 'rigid' propositional functions.

abbreviation *lift-conv*::$('t{\Rightarrow}bool){\Rightarrow}('t{\Rightarrow}\sigma)$ ($(\!|$-$|\!)$) **where** $(\!|S|\!) \equiv \lambda X.\ \lambda w.\ S\ X$

We introduce an useful operator: the range of a propositional function (resp. restricted over a domain),

definition *pfunRange*::$('t{\Rightarrow}\sigma){\Rightarrow}(\sigma{\Rightarrow}bool)$ (Ra(-)) **where** $Ra(\pi) \equiv \lambda Y.\ \exists x.\ (\pi\ x) = Y$
definition *pfunRange-restr*::$('t{\Rightarrow}\sigma){\Rightarrow}('t{\Rightarrow}bool){\Rightarrow}(\sigma{\Rightarrow}bool)$ (Ra[-|-]) **where** $Ra[\pi|D] \equiv \lambda Y.\ \exists x.\ (D\ x) \wedge (\pi\ x) = Y$

and check that taking infinite joins/meets (suprema/infima) over the range of a propositional function can be equivalently codified by using quantifiers. This is a quite useful simplifying relationship.

lemma *Ra-all*: $\bigwedge Ra(\pi) = \forall\,\pi$ ⟨*proof*⟩
lemma *Ra-ex*: $\bigvee Ra(\pi) = \exists\,\pi$ ⟨*proof*⟩
lemma *Ra-restr-all*: $\bigwedge Ra[\pi|D] = \forall^R(\!|D|\!)\pi$ ⟨*proof*⟩
lemma *Ra-restr-ex*: $\bigvee Ra[\pi|D] = \exists^R(\!|D|\!)\pi$ ⟨*proof*⟩

We further introduce the positive (negative) restriction of a propositional function wrt. a domain,

abbreviation *pfunRestr-pos*::$('t{\Rightarrow}\sigma){\Rightarrow}('t{\Rightarrow}\sigma){\Rightarrow}('t{\Rightarrow}\sigma)$ ($[\text{-}|\text{-}]^P$) **where** $[\pi|\delta]^P \equiv \lambda X.\ \lambda w.\ (\delta\ X)\ w \longrightarrow (\pi\ X)\ w$
abbreviation *pfunRestr-neg*::$('t{\Rightarrow}\sigma){\Rightarrow}('t{\Rightarrow}\sigma){\Rightarrow}('t{\Rightarrow}\sigma)$ ($[\text{-}|\text{-}]^N$) **where** $[\pi|\delta]^N \equiv \lambda X.\ \lambda w.\ (\delta\ X)\ w\ \wedge\ (\pi\ X)\ w$

and check that some additional simplifying relationships obtain.

lemma *all-restr*: $\forall^R(\delta)\pi = \forall\,[\pi|\delta]^P$ ⟨*proof*⟩
lemma *ex-restr*: $\exists^R(\delta)\pi = \exists\,[\pi|\delta]^N$ ⟨*proof*⟩
lemma *Ra-all-restr*: $\bigwedge Ra[\pi|D] = \forall\,[\pi|(\!|D|\!)]^P$ ⟨*proof*⟩
lemma *Ra-ex-restr*: $\bigvee Ra[\pi|D] = \exists\,[\pi|(\!|D|\!)]^N$ ⟨*proof*⟩

Observe that using these operators has the advantage of allowing for binder notation,

lemma $\forall X.\ [\pi|\delta]^P\ X = \forall\,[\pi|\delta]^P$ ⟨*proof*⟩
lemma $\exists X.\ [\pi|\delta]^N\ X = \exists\,[\pi|\delta]^N$ ⟨*proof*⟩

noting that extra care should be taken when working with complements or negations; always remember to switch P/N (positive/negative restriction) accordingly.

lemma $\forall^R(\delta)\pi\ = \forall X.\ \ [\pi|\delta]^P\ X$ ⟨*proof*⟩
lemma $\forall^R(\delta)\pi^c = \forall X.\ -[\pi|\delta]^N\ X$ ⟨*proof*⟩
lemma $\exists^R(\delta)\pi\ = \exists X.\ \ [\pi|\delta]^N\ X$ ⟨*proof*⟩
lemma $\exists^R(\delta)\pi^c = \exists X.\ -[\pi|\delta]^P\ X$ ⟨*proof*⟩

The previous definitions allow us to nicely characterize the interaction between function composition and (restricted) quantification:

lemma *Ra-all-comp1*: $\forall\,(\pi{\circ}\gamma) = \forall\,[\pi|(\!|Ra\ \gamma|\!)]^P$ ⟨*proof*⟩

lemma *Ra-all-comp2*: $\forall\,(\pi\circ\gamma) = \forall^R(\!|Ra\ \gamma|\!)\ \pi$ $\langle proof\rangle$
lemma *Ra-ex-comp1*: $\exists\,(\pi\circ\gamma) = \exists\,[\pi|(\!|Ra\ \gamma|\!)]^N$ $\langle proof\rangle$
lemma *Ra-ex-comp2*: $\exists\,(\pi\circ\gamma) = \exists^R(\!|Ra\ \gamma|\!)\ \pi$ $\langle proof\rangle$

This useful operator returns for a given domain of propositions the domain of their complements:

definition *dom-compl*::$(\sigma\Rightarrow bool)\Rightarrow(\sigma\Rightarrow bool)$ $((\text{-}^{-1}))$ **where** $D^{-1} \equiv \lambda X.\ \exists\,Y.\ (D\ Y) \wedge (X = -Y)$
lemma *dom-compl-def2*: $D^{-1} = (\lambda X.\ D(-X))$ $\langle proof\rangle$
lemma *dom-compl-invol*: $D = (D^{-1})^{-1}$ $\langle proof\rangle$

We can now check an infinite variant of the De Morgan laws,

lemma *iDM-a*: $-(\bigwedge S) = \bigvee S^{-1}$ $\langle proof\rangle$
lemma *iDM-b*: $-(\bigvee S) = \bigwedge S^{-1}$ $\langle proof\rangle$

and some useful dualities regarding the range of propositional functions (restricted wrt. a domain).

lemma *Ra-compl*: $Ra[\pi^c|D] \ = Ra[\pi|D]^{-1}$ $\langle proof\rangle$
lemma *Ra-dual1*: $Ra[\pi^d|D] \ = Ra[\pi|D^{-1}]^{-1}$ $\langle proof\rangle$
lemma *Ra-dual2*: $Ra[\pi^d|D] \ = Ra[\pi^c|D^{-1}]$ $\langle proof\rangle$
lemma *Ra-dual3*: $Ra[\pi^d|D]^{-1} = Ra[\pi|D^{-1}]$ $\langle proof\rangle$
lemma *Ra-dual4*: $Ra[\pi^d|D^{-1}] = Ra[\pi|D]^{-1}$ $\langle proof\rangle$

Finally, we check some facts concerning duality for quantifiers.

lemma $\exists\,\pi^c = -(\forall\,\pi)$ $\langle proof\rangle$
lemma $\forall\,\pi^c = -(\exists\,\pi)$ $\langle proof\rangle$
lemma $\exists\,X.\ -\pi\ X = -(\forall\,X.\ \pi\ X)$ $\langle proof\rangle$
lemma $\forall\,X.\ -\pi\ X = -(\exists\,X.\ \pi\ X)$ $\langle proof\rangle$

lemma $\exists^R(\delta)\pi^c = -(\forall^R(\delta)\pi)$ $\langle proof\rangle$
lemma $\forall^R(\delta)\pi^c = -(\exists^R(\delta)\pi)$ $\langle proof\rangle$
lemma $\exists\,X.\ -[\pi|\delta]^P\ X = -(\forall\,X.\ [\pi|\delta]^P\ X)$ $\langle proof\rangle$
lemma $\forall\,X.\ -[\pi|\delta]^P\ X = -(\exists\,X.\ [\pi|\delta]^P\ X)$ $\langle proof\rangle$
lemma $\exists\,X.\ -[\pi|\delta]^N\ X = -(\forall\,X.\ [\pi|\delta]^N\ X)$ $\langle proof\rangle$
lemma $\forall\,X.\ -[\pi|\delta]^N\ X = -(\exists\,X.\ [\pi|\delta]^N\ X)$ $\langle proof\rangle$

Warning: Do not switch P and N when passing to the dual form.

lemma $\forall\,X.\ [\pi|\delta]^P\ X = -(\exists\,X.\ -[\pi|\delta]^N\ X)$ **nitpick** $\langle proof\rangle$
lemma $\forall\,X.\ [\pi|\delta]^P\ X = -(\exists\,X.\ -[\pi|\delta]^P\ X)$ $\langle proof\rangle$

end
theory *sse-operation-positive*
  imports *sse-boolean-algebra*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

2 Positive semantic conditions for operations

We define and interrelate some useful conditions on propositional functions which do not involve negative-like properties (hence 'positive'). We focus on propositional functions which correspond to unary connectives of the algebra (with type $\sigma\Rightarrow\sigma$). We call such propositional functions 'operations'.

2.1 Definitions (finitary case)

Monotonicity (MONO).

definition *MONO* $\varphi \equiv \forall A\ B.\ A \preceq B \longrightarrow \varphi\ A \preceq \varphi\ B$
lemma *MONO-ant*: $MONO\ \varphi \Longrightarrow \forall A\ B\ C.\ A \preceq B \longrightarrow \varphi(B \rightarrow C) \preceq \varphi(A \rightarrow C)$ ⟨*proof*⟩
lemma *MONO-cons*: $MONO\ \varphi \Longrightarrow \forall A\ B\ C.\ A \preceq B \longrightarrow \varphi(C \rightarrow A) \preceq \varphi(C \rightarrow B)$ ⟨*proof*⟩
lemma *MONO-dual*: $MONO\ \varphi \Longrightarrow MONO\ \varphi^d$ ⟨*proof*⟩

Extensive/expansive (EXP) and its dual (dEXP), aka. 'contractive'.

definition *EXP* $\varphi \equiv \forall A.\ A \preceq \varphi\ A$
definition *dEXP* $\varphi \equiv \forall A.\ \varphi\ A \preceq A$
lemma *EXP-dual1*: $EXP\ \varphi \Longrightarrow dEXP\ \varphi^d$ ⟨*proof*⟩
lemma *EXP-dual2*: $dEXP\ \varphi \Longrightarrow EXP\ \varphi^d$ ⟨*proof*⟩

Idempotence (IDEM).

definition *IDEM* $\varphi \equiv \forall A.\ (\varphi\ A) \approx \varphi(\varphi\ A)$
definition *IDEMa* $\varphi \equiv \forall A.\ (\varphi\ A) \preceq \varphi(\varphi\ A)$
definition *IDEMb* $\varphi \equiv \forall A.\ (\varphi\ A) \succeq \varphi(\varphi\ A)$
lemma *IDEM-dual1*: $IDEMa\ \varphi \Longrightarrow IDEMb\ \varphi^d$ ⟨*proof*⟩
lemma *IDEM-dual2*: $IDEMb\ \varphi \Longrightarrow IDEMa\ \varphi^d$ ⟨*proof*⟩
lemma *IDEM-dual*: $IDEM\ \varphi = IDEM\ \varphi^d$ ⟨*proof*⟩

Normality (NOR) and its dual (dNOR).

definition *NOR* $\varphi \equiv (\varphi\ \bot) \approx \bot$
definition *dNOR* $\varphi \equiv (\varphi\ \top) \approx \top$
lemma *NOR-dual1*: $NOR\ \varphi = dNOR\ \varphi^d$ ⟨*proof*⟩
lemma *NOR-dual2*: $dNOR\ \varphi = NOR\ \varphi^d$ ⟨*proof*⟩

Distribution over meets or multiplicativity (MULT).

definition *MULT* $\varphi \equiv \forall A\ B.\ \varphi(A \wedge B) \approx (\varphi\ A) \wedge (\varphi\ B)$
definition *MULT-a* $\varphi \equiv \forall A\ B.\ \varphi(A \wedge B) \preceq (\varphi\ A) \wedge (\varphi\ B)$
definition *MULT-b* $\varphi \equiv \forall A\ B.\ \varphi(A \wedge B) \succeq (\varphi\ A) \wedge (\varphi\ B)$

Distribution over joins or additivity (ADDI).

definition *ADDI* $\varphi \equiv \forall A\ B.\ \varphi(A \vee B) \approx (\varphi\ A) \vee (\varphi\ B)$
definition *ADDI-a* $\varphi \equiv \forall A\ B.\ \varphi(A \vee B) \preceq (\varphi\ A) \vee (\varphi\ B)$
definition *ADDI-b* $\varphi \equiv \forall A\ B.\ \varphi(A \vee B) \succeq (\varphi\ A) \vee (\varphi\ B)$

2.2 Relations among conditions (finitary case)

dEXP and dNOR entail NOR.

lemma $dEXP\ \varphi \Longrightarrow dNOR\ \varphi \Longrightarrow NOR\ \varphi$ ⟨*proof*⟩

EXP and NOR entail dNOR.

lemma $EXP\ \varphi \Longrightarrow NOR\ \varphi \Longrightarrow dNOR\ \varphi$ ⟨*proof*⟩

Interestingly, EXP and its dual allow for an alternative characterization of fixed-point operators.

lemma *EXP-fp*: $EXP\ \varphi \Longrightarrow \varphi^{fp} \equiv (\varphi^c \sqcup id)$ ⟨*proof*⟩
lemma *dEXP-fp*: $dEXP\ \varphi \Longrightarrow \varphi^{fp} \equiv (\varphi \sqcup compl)$ ⟨*proof*⟩

MONO, MULT-a and ADDI-b are equivalent.

lemma *MONO-MULTa*: $MONO\ \varphi = MULT\text{-}a\ \varphi$ ⟨*proof*⟩

lemma *MONO-ADDIb*: $MONO\ \varphi = ADDI\text{-}b\ \varphi$ ⟨*proof*⟩
lemma *ADDIb-MULTa*: $ADDI\text{-}b\ \varphi = MULT\text{-}a\ \varphi$ ⟨*proof*⟩

end
theory *sse-operation-positive-quantification*
imports *sse-operation-positive sse-boolean-algebra-quantification*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

2.3 Definitions (infinitary case)

We define and interrelate infinitary variants for some previously introduced ('positive') conditions on operations and show how they relate to quantifiers as previously defined.

Distribution over infinite meets (infima) or infinite multiplicativity (iMULT).

definition $iMULT\ \varphi \equiv \forall S.\ \varphi(\bigwedge S) \approx \bigwedge Ra[\varphi|S]$
definition $iMULT\text{-}a\ \varphi \equiv \forall S.\ \varphi(\bigwedge S) \preceq \bigwedge Ra[\varphi|S]$
definition $iMULT\text{-}b\ \varphi \equiv \forall S.\ \varphi(\bigwedge S) \succeq \bigwedge Ra[\varphi|S]$

Distribution over infinite joins (suprema) or infinite additivity (iADDI).

definition $iADDI\ \varphi \equiv \forall S.\ \varphi(\bigvee S) \approx \bigvee Ra[\varphi|S]$
definition $iADDI\text{-}a\ \varphi \equiv \forall S.\ \varphi(\bigvee S) \preceq \bigvee Ra[\varphi|S]$
definition $iADDI\text{-}b\ \varphi \equiv \forall S.\ \varphi(\bigvee S) \succeq \bigvee Ra[\varphi|S]$

2.4 Relations among conditions (infinitary case)

We start by noting that there is a duality between iADDI-a and iMULT-b.

lemma *iADDI-MULT-dual1*: $iADDI\text{-}a\ \varphi \Longrightarrow iMULT\text{-}b\ \varphi^d$ ⟨*proof*⟩
lemma *iADDI-MULT-dual2*: $iMULT\text{-}b\ \varphi \Longrightarrow iADDI\text{-}a\ \varphi^d$ ⟨*proof*⟩

MULT-a and iMULT-a are equivalent.

lemma *iMULTa-rel*: $iMULT\text{-}a\ \varphi = MULT\text{-}a\ \varphi$ ⟨*proof*⟩

ADDI-b and iADDI-b are equivalent.

lemma *iADDIb-rel*: $iADDI\text{-}b\ \varphi = ADDI\text{-}b\ \varphi$ ⟨*proof*⟩

Thus we have that MONO, MULT-a/iMULT-a and ADDI-b/iADDI-b are all equivalent.

lemma *MONO-iADDIb*: $MONO\ \varphi = iADDI\text{-}b\ \varphi$ ⟨*proof*⟩
lemma *MONO-iMULTa*: $MONO\ \varphi = iMULT\text{-}a\ \varphi$ ⟨*proof*⟩
lemma *iADDI-b-iMULTa*: $iADDI\text{-}b\ \varphi = iMULT\text{-}a\ \varphi$ ⟨*proof*⟩

lemma *PI-imult*: $MONO\ \varphi \Longrightarrow iMULT\text{-}b\ \varphi \Longrightarrow iMULT\ \varphi$ ⟨*proof*⟩
lemma *PC-iaddi*: $MONO\ \varphi \Longrightarrow iADDI\text{-}a\ \varphi \Longrightarrow iADDI\ \varphi$ ⟨*proof*⟩

Interestingly, we can show that suitable (infinitary) conditions on an operation can make the set of its fixed points closed under infinite meets/joins.

lemma *fp-inf-closed*: $MONO\ \varphi \Longrightarrow iMULT\text{-}b\ \varphi \Longrightarrow$ *infimum-closed* (*fp* φ) ⟨*proof*⟩
lemma *fp-sup-closed*: $MONO\ \varphi \Longrightarrow iADDI\text{-}a\ \varphi \Longrightarrow$ *supremum-closed* (*fp* φ) ⟨*proof*⟩

2.5 Exploring the Barcan formula and its converse

The converse Barcan formula follows readily from monotonicity.

lemma *CBarcan1*: $MONO\ \varphi \Longrightarrow \forall \pi.\ \ \varphi(\forall x.\ \pi\ x)\ \preceq (\forall x.\ \varphi(\pi\ x))$ ⟨*proof*⟩
lemma *CBarcan2*: $MONO\ \varphi \Longrightarrow \forall \pi.\ (\exists x.\ \varphi(\pi\ x)) \preceq \varphi(\exists x.\ \pi\ x)$ ⟨*proof*⟩

However, the Barcan formula requires a stronger assumption (of an infinitary character).

lemma *Barcan1*: $iMULT\text{-}b\ \varphi \Longrightarrow \forall \pi.\ (\forall x.\ \varphi(\pi\ x)) \preceq \varphi(\forall x.\ \pi\ x)$ ⟨*proof*⟩
lemma *Barcan2*: $iADDI\text{-}a\ \varphi \Longrightarrow \forall \pi.\ \varphi(\exists x.\ \pi\ x) \preceq (\exists x.\ \varphi(\pi\ x))$ ⟨*proof*⟩

end
theory *sse-operation-negative*
  imports *sse-boolean-algebra*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

3 Negative semantic conditions for operations

We define and interrelate some conditions on operations (i.e. propositional functions of type $\sigma{\Rightarrow}\sigma$), this time involving negative-like properties.

named-theorems *Defs*

3.1 Definitions and interrelations (finitary case)

3.1.1 Principles of excluded middle, contradiction and explosion

TND: tertium non datur, aka. law of excluded middle (resp. strong, weak, minimal).

abbreviation *pTND* (TND^{-} -) **where** $TND^{a}\ \ \eta \equiv\ \ \ \ \ \top\ \approx a \vee (\eta\ a)$
abbreviation *pTNDw* ($TNDw^{-}$ -) **where** $TNDw^{a}\ \eta \equiv \forall b.\ (\eta\ b) \preceq a \vee (\eta\ a)$
abbreviation *pTNDm* ($TNDm^{-}$ -) **where** $TNDm^{a}\ \eta \equiv\ \ \ \ (\eta\ \bot) \preceq a \vee (\eta\ a)$
definition $TND\ \ \eta \equiv \forall \varphi.\ TND^{\varphi}\ \ \eta$
definition $TNDw\ \eta \equiv \forall \varphi.\ TNDw^{\varphi}\ \eta$
definition $TNDm\ \eta \equiv \forall \varphi.\ TNDm^{\varphi}\ \eta$
declare *TND-def*[*Defs*] *TNDw-def*[*Defs*] *TNDm-def*[*Defs*]

Explore some (non)entailment relations:

lemma $TND\ \ \eta \Longrightarrow TNDw\ \eta$ ⟨*proof*⟩
lemma $TNDw\ \eta \Longrightarrow TND\ \ \eta$ **nitpick** ⟨*proof*⟩
lemma $TNDw\ \eta \Longrightarrow TNDm\ \eta$ ⟨*proof*⟩
lemma $TNDm\ \eta \Longrightarrow TNDw\ \eta$ **nitpick** ⟨*proof*⟩

ECQ: ex contradictione (sequitur) quodlibet (resp: strong, weak, minimal).

abbreviation *pECQ* (ECQ^{-} -) **where** $ECQ^{a}\ \ \eta \equiv\ \ \ \ a \wedge (\eta\ a) \approx \bot$
abbreviation *pECQw* ($ECQw^{-}$ -) **where** $ECQw^{a}\ \eta \equiv \forall b.\ a \wedge (\eta\ a) \preceq (\eta\ b)$
abbreviation *pECQm* ($ECQm^{-}$ -) **where** $ECQm^{a}\ \eta \equiv\ \ \ \ a \wedge (\eta\ a) \preceq (\eta\ \top)$
definition $ECQ\ \ \eta \equiv \forall a.\ ECQ^{a}\ \ \eta$
definition $ECQw\ \eta \equiv \forall a.\ ECQw^{a}\ \eta$
definition $ECQm\ \eta \equiv \forall a.\ ECQm^{a}\ \eta$
declare *ECQ-def*[*Defs*] *ECQw-def*[*Defs*] *ECQm-def*[*Defs*]

Explore some (non)entailment relations:

lemma $ECQ\ \ \eta \Longrightarrow ECQw\ \eta$ ⟨*proof*⟩
lemma $ECQw\ \eta \Longrightarrow ECQ\ \ \eta$ **nitpick** ⟨*proof*⟩
lemma $ECQw\ \eta \Longrightarrow ECQm\ \eta$ ⟨*proof*⟩
lemma $ECQm\ \eta \Longrightarrow ECQw\ \eta$ **nitpick** ⟨*proof*⟩

LNC: law of non-contradiction.

abbreviation $pLNC$ $(LNC^{-}\ \text{-})$ **where** $LNC^{a}\ \eta \equiv \eta(a \wedge \eta\ a) \approx \top$
definition $LNC\ \eta \equiv \forall\, a.\ LNC^{a}\ \eta$
declare $LNC\text{-}def[Defs]$

ECQ and LNC are in general independent.

lemma $ECQ\ \eta \Longrightarrow LNC\ \eta$ **nitpick** $\langle proof \rangle$
lemma $LNC\ \eta \Longrightarrow ECQm\ \eta$ **nitpick** $\langle proof \rangle$

3.1.2 Contraposition rules

CoP: contraposition (global/rule variants, resp. weak, strong var. 1, strong var. 2, strong var. 3).

abbreviation $pCoPw$ $(CoPw^{--}\ \text{-})$ **where** $CoPw^{ab}\ \eta \equiv a \preceq b \longrightarrow (\eta\ b) \preceq (\eta\ a)$
abbreviation $pCoP1$ $(CoP1^{--}\ \text{-})$ **where** $CoP1^{ab}\ \eta \equiv a \preceq (\eta\ b) \longrightarrow b \preceq (\eta\ a)$
abbreviation $pCoP2$ $(CoP2^{--}\ \text{-})$ **where** $CoP2^{ab}\ \eta \equiv (\eta\ a) \preceq b \longrightarrow (\eta\ b) \preceq a$
abbreviation $pCoP3$ $(CoP3^{--}\ \text{-})$ **where** $CoP3^{ab}\ \eta \equiv (\eta\ a) \preceq (\eta\ b) \longrightarrow b \preceq a$
definition $CoPw\ \ \eta \equiv \forall\, a\ b.\ CoPw^{ab}\ \eta$
definition $CoP1\ \ \eta \equiv \forall\, a\ b.\ CoP1^{ab}\ \eta$
definition $CoP1'\ \eta \equiv \forall\, a\ b.\ a \preceq (\eta\ b) \longleftrightarrow b \preceq (\eta\ a)$
definition $CoP2\ \ \eta \equiv \forall\, a\ b.\ CoP2^{ab}\ \eta$
definition $CoP2'\ \eta \equiv \forall\, a\ b.\ (\eta\ a) \preceq b \longleftrightarrow (\eta\ b) \preceq a$
definition $CoP3\ \ \eta \equiv \forall\, a\ b.\ CoP3^{ab}\ \eta$
declare $CoPw\text{-}def[Defs]$ $CoP1\text{-}def[Defs]$ $CoP1'\text{-}def[Defs]$
$CoP2\text{-}def[Defs]$ $CoP2'\text{-}def[Defs]$ $CoP3\text{-}def[Defs]$

lemma $CoP1\text{-}defs\text{-}rel$: $CoP1\ \eta = CoP1'\ \eta$ $\langle proof \rangle$
lemma $CoP2\text{-}defs\text{-}rel$: $CoP2\ \eta = CoP2'\ \eta$ $\langle proof \rangle$

Explore some (non)entailment relations:

lemma $CoP1\ \eta \Longrightarrow CoPw\ \eta$ $\langle proof \rangle$
lemma $CoPw\ \eta \Longrightarrow CoP1\ \eta$ **nitpick** $\langle proof \rangle$
lemma $CoP2\ \eta \Longrightarrow CoPw\ \eta$ $\langle proof \rangle$
lemma $CoPw\ \eta \Longrightarrow CoP2\ \eta$ **nitpick** $\langle proof \rangle$
lemma $CoP3\ \eta \Longrightarrow CoPw\ \eta$ $\langle proof \rangle$
lemma $CoPw\ \eta \Longrightarrow CoP3\ \eta$ **nitpick** $\langle proof \rangle$

All three strong variants are pairwise independent. However, CoP3 follows from CoP1 plus CoP2.

lemma $CoP123$: $CoP1\ \eta \Longrightarrow CoP2\ \eta \Longrightarrow CoP3\ \eta$ $\langle proof \rangle$

Taking all CoP together still leaves room for a boldly paraconsistent resp. paracomplete logic.

lemma $CoP1\ \eta \Longrightarrow CoP2\ \eta \Longrightarrow ECQm\ \eta$ **nitpick** $\langle proof \rangle$
lemma $CoP1\ \eta \Longrightarrow CoP2\ \eta \Longrightarrow TNDm\ \eta$ **nitpick** $\langle proof \rangle$

3.1.3 Modus tollens rules

MT: modus (tollendo) tollens (global/rule variants).

abbreviation $pMT0$ $(MT0^{--}\ \text{-})$ **where** $MT0^{ab}\ \eta \equiv a \preceq b \wedge (\eta\ b) \approx \top \longrightarrow (\eta\ a) \approx \top$
abbreviation $pMT1$ $(MT1^{--}\ \text{-})$ **where** $MT1^{ab}\ \eta \equiv a \preceq (\eta\ b) \wedge b \approx \top \longrightarrow (\eta\ a) \approx \top$
abbreviation $pMT2$ $(MT2^{--}\ \text{-})$ **where** $MT2^{ab}\ \eta \equiv (\eta\ a) \preceq b \wedge (\eta\ b) \approx \top \longrightarrow a \approx \top$
abbreviation $pMT3$ $(MT3^{--}\ \text{-})$ **where** $MT3^{ab}\ \eta \equiv (\eta\ a) \preceq (\eta\ b) \wedge b \approx \top \longrightarrow a \approx \top$
definition $MT0\ \eta \equiv \forall\, a\ b.\ MT0^{ab}\ \eta$

definition $MT1\ \eta \equiv \forall\, a\ b.\ MT1^{a\,b}\ \eta$
definition $MT2\ \eta \equiv \forall\, a\ b.\ MT2^{a\,b}\ \eta$
definition $MT3\ \eta \equiv \forall\, a\ b.\ MT3^{a\,b}\ \eta$
declare *MT0-def*[*Defs*] *MT1-def*[*Defs*] *MT2-def*[*Defs*] *MT3-def*[*Defs*]

Again, all MT variants are pairwise independent. We explore some (non)entailment relations:

lemma $CoPw\ \eta \Longrightarrow MT0\ \eta$ ⟨*proof*⟩
lemma $CoP1\ \eta \Longrightarrow MT1\ \eta$ ⟨*proof*⟩
lemma $CoP2\ \eta \Longrightarrow MT2\ \eta$ ⟨*proof*⟩
lemma $CoP3\ \eta \Longrightarrow MT3\ \eta$ ⟨*proof*⟩
lemma $MT0\ \eta \Longrightarrow MT1\ \eta \Longrightarrow MT2\ \eta \Longrightarrow MT3\ \eta \Longrightarrow CoPw\ \eta$ **nitpick** ⟨*proof*⟩
lemma $MT0\ \eta \Longrightarrow MT1\ \eta \Longrightarrow MT2\ \eta \Longrightarrow MT3\ \eta \Longrightarrow ECQm\ \eta$ **nitpick** ⟨*proof*⟩
lemma $MT0\ \eta \Longrightarrow MT1\ \eta \Longrightarrow MT2\ \eta \Longrightarrow MT3\ \eta \Longrightarrow TNDm\ \eta$ **nitpick** ⟨*proof*⟩
lemma *MT123*: $MT1\ \eta \Longrightarrow MT2\ \eta \Longrightarrow MT3\ \eta$ ⟨*proof*⟩

3.1.4 Double negation introduction and elimination

DNI/DNE: double negation introduction/elimination (as axioms).

abbreviation *pDNI* (DNI^{-} -) **where** $DNI^{a}\ \eta \equiv a \preceq \eta\ (\eta\ a)$
abbreviation *pDNE* (DNE^{-} -) **where** $DNE^{a}\ \eta \equiv \eta\ (\eta\ a) \preceq a$
definition $DNI\ \eta \equiv \forall\, a.\ DNI^{a}\ \eta$
definition $DNE\ \eta \equiv \forall\, a.\ DNE^{a}\ \eta$
declare *DNI-def*[*Defs*] *DNE-def*[*Defs*]

CoP1 (resp. CoP2) can alternatively be defined as CoPw plus DNI (resp. DNE).

lemma $DNI\ \eta \Longrightarrow CoP1\ \eta$ **nitpick** ⟨*proof*⟩
lemma *CoP1-def2*: $CoP1\ \eta = (CoPw\ \eta \wedge DNI\ \eta)$ ⟨*proof*⟩
lemma $DNE\ \eta \Longrightarrow CoP2\ \eta$ **nitpick** ⟨*proof*⟩
lemma *CoP2-def2*: $CoP2\ \eta = (CoPw\ \eta \wedge DNE\ \eta)$ ⟨*proof*⟩

Explore some non-entailment relations:

lemma $DNI\ \eta \Longrightarrow DNE\ \eta \Longrightarrow CoPw\ \eta$ **nitpick** ⟨*proof*⟩
lemma $DNI\ \eta \Longrightarrow DNE\ \eta \Longrightarrow TNDm\ \eta$ **nitpick** ⟨*proof*⟩
lemma $DNI\ \eta \Longrightarrow DNE\ \eta \Longrightarrow ECQm\ \eta$ **nitpick** ⟨*proof*⟩
lemma $DNI\ \eta \Longrightarrow DNE\ \eta \Longrightarrow MT0\ \eta$ **nitpick** ⟨*proof*⟩
lemma $DNI\ \eta \Longrightarrow DNE\ \eta \Longrightarrow MT1\ \eta$ **nitpick** ⟨*proof*⟩
lemma $DNI\ \eta \Longrightarrow DNE\ \eta \Longrightarrow MT2\ \eta$ **nitpick** ⟨*proof*⟩
lemma $DNI\ \eta \Longrightarrow DNE\ \eta \Longrightarrow MT3\ \eta$ **nitpick** ⟨*proof*⟩

DNI/DNE: double negation introduction/elimination (as rules).

abbreviation *prDNI* ($rDNI^{-}$ -) **where** $rDNI^{a}\ \eta \equiv a \approx \top \longrightarrow \eta\ (\eta\ a) \approx \top$
abbreviation *prDNE* ($rDNE^{-}$ -) **where** $rDNE^{a}\ \eta \equiv \eta\ (\eta\ a) \approx \top \longrightarrow a \approx \top$
definition $rDNI\ \eta \equiv \forall\, a.\ rDNI^{a}\ \eta$
definition $rDNE\ \eta \equiv \forall\, a.\ rDNE^{a}\ \eta$
declare *rDNI-def*[*Defs*] *rDNE-def*[*Defs*]

The rule variants are strictly weaker than the axiom variants,

lemma $DNI\ \eta \Longrightarrow rDNI\ \eta$ ⟨*proof*⟩
lemma $rDNI\ \eta \Longrightarrow DNI\ \eta$ **nitpick** ⟨*proof*⟩
lemma $DNE\ \eta \Longrightarrow rDNE\ \eta$ ⟨*proof*⟩
lemma $rDNE\ \eta \Longrightarrow DNE\ \eta$ **nitpick** ⟨*proof*⟩

and follow already from modus tollens.

lemma *MT1-rDNI*: $MT1\ \eta \Longrightarrow rDNI\ \eta$ ⟨*proof*⟩
lemma *MT2-rDNE*: $MT2\ \eta \Longrightarrow rDNE\ \eta$ ⟨*proof*⟩

3.1.5 Normality and its dual

n(D)Nor: negative (dual) 'normality'.

definition *nNor* $\eta \equiv (\eta\ \bot) \approx \top$
definition *nDNor* $\eta \equiv (\eta\ \top) \approx \bot$
declare *nNor-def[Defs] nDNor-def[Defs]*

nNor (resp. nDNor) is entailed by CoP1 (resp. CoP2).

lemma *CoP1-Nor*: *CoP1* $\eta \Longrightarrow$ *nNor* η ⟨*proof*⟩
lemma *CoP2-DNor*: *CoP2* $\eta \Longrightarrow$ *nDNor* η ⟨*proof*⟩
lemma *DNI* $\eta \Longrightarrow$ *nNor* η **nitpick** ⟨*proof*⟩
lemma *DNE* $\eta \Longrightarrow$ *nDNor* η **nitpick** ⟨*proof*⟩

nNor and nDNor together entail the rule variant of DNI (rDNI).

lemma *nDNor-rDNI*: *nNor* $\eta \Longrightarrow$ *nDNor* $\eta \Longrightarrow$ *rDNI* η ⟨*proof*⟩
lemma *nNor* $\eta \Longrightarrow$ *nDNor* $\eta \Longrightarrow$ *rDNE* η **nitpick** ⟨*proof*⟩

3.1.6 De Morgan laws

DM: De Morgan laws.

abbreviation *pDM1* (*DM1*$^{--}$ -) **where** *DM1*^{ab} $\eta \equiv \eta(a \vee b) \preceq (\eta\ a) \wedge (\eta\ b)$
abbreviation *pDM2* (*DM2*$^{--}$ -) **where** *DM2*^{ab} $\eta \equiv (\eta\ a) \vee (\eta\ b) \preceq \eta(a \wedge b)$
abbreviation *pDM3* (*DM3*$^{--}$ -) **where** *DM3*^{ab} $\eta \equiv \eta(a \wedge b) \preceq (\eta\ a) \vee (\eta\ b)$
abbreviation *pDM4* (*DM4*$^{--}$ -) **where** *DM4*^{ab} $\eta \equiv (\eta\ a) \wedge (\eta\ b) \preceq \eta(a \vee b)$
definition *DM1* $\eta \equiv \forall\, a\ b.$ *DM1*^{ab} η
definition *DM2* $\eta \equiv \forall\, a\ b.$ *DM2*^{ab} η
definition *DM3* $\eta \equiv \forall\, a\ b.$ *DM3*^{ab} η
definition *DM4* $\eta \equiv \forall\, a\ b.$ *DM4*^{ab} η
declare *DM1-def[Defs] DM2-def[Defs] DM3-def[Defs] DM4-def[Defs]*

CoPw, DM1 and DM2 are indeed equivalent.

lemma *DM1-CoPw*: *DM1* $\eta =$ *CoPw* η ⟨*proof*⟩
lemma *DM2-CoPw*: *DM2* $\eta =$ *CoPw* η ⟨*proof*⟩
lemma *DM12*: *DM1* $\eta =$ *DM2* η ⟨*proof*⟩

DM3 (resp. DM4) are entailed by CoPw together with DNE (resp. DNI).

lemma *CoPw-DNE-DM3*: *CoPw* $\eta \Longrightarrow$ *DNE* $\eta \Longrightarrow$ *DM3* η ⟨*proof*⟩
lemma *CoPw-DNI-DM4*: *CoPw* $\eta \Longrightarrow$ *DNI* $\eta \Longrightarrow$ *DM4* η ⟨*proof*⟩

From this follows that DM3 (resp. DM4) is entailed by CoP2 (resp. CoP1).

lemma *CoP2-DM3*: *CoP2* $\eta \Longrightarrow$ *DM3* η ⟨*proof*⟩
lemma *CoP1-DM4*: *CoP1* $\eta \Longrightarrow$ *DM4* η ⟨*proof*⟩

Explore some non-entailment relations:

lemma *CoPw* $\eta \Longrightarrow$ *DM3* $\eta \Longrightarrow$ *DM4* $\eta \Longrightarrow$ *nNor* $\eta \Longrightarrow$ *nDNor* $\eta \Longrightarrow$ *DNI* η **nitpick** ⟨*proof*⟩
lemma *CoPw* $\eta \Longrightarrow$ *DM3* $\eta \Longrightarrow$ *DM4* $\eta \Longrightarrow$ *nNor* $\eta \Longrightarrow$ *nDNor* $\eta \Longrightarrow$ *DNE* η **nitpick** ⟨*proof*⟩
lemma *CoPw* $\eta \Longrightarrow$ *DM3* $\eta \Longrightarrow$ *DM4* $\eta \Longrightarrow$ *DNI* $\eta \Longrightarrow$ *DNE* $\eta \Longrightarrow$ *ECQm* η **nitpick** ⟨*proof*⟩
lemma *CoPw* $\eta \Longrightarrow$ *DM3* $\eta \Longrightarrow$ *DM4* $\eta \Longrightarrow$ *DNI* $\eta \Longrightarrow$ *DNE* $\eta \Longrightarrow$ *TNDm* η **nitpick** ⟨*proof*⟩

3.1.7 Contextual (strong) contraposition rule

XCoP: contextual contraposition (global/rule variant).

abbreviation *pXCoP* (*XCoP*$^{\text{--}}$ -) **where** $XCoP^{ab}\ \eta \equiv \forall\, c.\ c \wedge a \preceq b \longrightarrow c \wedge (\eta\ b) \preceq (\eta\ a)$
definition $XCoP\ \eta \equiv \forall\, a\ b.\ XCoP^{ab}\ \eta$
declare *XCoP-def*[*Defs*]

XCoP can alternatively be defined as ECQw plus TNDw.

lemma *XCoP-def2*: $XCoP\ \eta = (ECQw\ \eta \wedge TNDw\ \eta)$ ⟨*proof*⟩

Explore some (non)entailment relations:

lemma $XCoP\ \eta \Longrightarrow ECQ\ \eta$ **nitpick** ⟨*proof*⟩
lemma $XCoP\ \eta \Longrightarrow TND\ \eta$ **nitpick** ⟨*proof*⟩
lemma *XCoP-CoPw*: $XCoP\ \eta \Longrightarrow CoPw\ \eta$ ⟨*proof*⟩
lemma $XCoP\ \eta \Longrightarrow CoP1\ \eta$ **nitpick** ⟨*proof*⟩
lemma $XCoP\ \eta \Longrightarrow CoP2\ \eta$ **nitpick** ⟨*proof*⟩
lemma $XCoP\ \eta \Longrightarrow CoP3\ \eta$ **nitpick** ⟨*proof*⟩
lemma $CoP1\ \eta \wedge CoP2\ \eta \Longrightarrow XCoP\ \eta$ **nitpick** ⟨*proof*⟩
lemma $XCoP\ \eta \Longrightarrow nNor\ \eta$ **nitpick** ⟨*proof*⟩
lemma $XCoP\ \eta \Longrightarrow nDNor\ \eta$ **nitpick** ⟨*proof*⟩
lemma $XCoP\ \eta \Longrightarrow rDNI\ \eta$ **nitpick** ⟨*proof*⟩
lemma $XCoP\ \eta \Longrightarrow rDNE\ \eta$ **nitpick** ⟨*proof*⟩
lemma *XCoP-DM3*: $XCoP\ \eta \Longrightarrow DM3\ \eta$ ⟨*proof*⟩
lemma *XCoP-DM4*: $XCoP\ \eta \Longrightarrow DM4\ \eta$ ⟨*proof*⟩

3.1.8 Local contraposition axioms

Observe that the definitions below take implication as an additional parameter: ι.

lCoP: contraposition (local/axiom variants).

abbreviation *plCoPw* (*lCoPw*$^{\text{--}}$ - -) **where** $lCoPw^{ab}\ \iota\ \eta \equiv (\iota\ a\ b{::}\sigma) \preceq (\iota\ (\eta\ b)\ (\eta\ a))$
abbreviation *plCoP1* (*lCoP1*$^{\text{--}}$ - -) **where** $lCoP1^{ab}\ \iota\ \eta \equiv (\iota\ a\ (\eta\ b{::}\sigma)) \preceq (\iota\ b\ (\eta\ a))$
abbreviation *plCoP2* (*lCoP2*$^{\text{--}}$ - -) **where** $lCoP2^{ab}\ \iota\ \eta \equiv (\iota\ (\eta\ a)\ b{::}\sigma) \preceq (\iota\ (\eta\ b)\ a)$
abbreviation *plCoP3* (*lCoP3*$^{\text{--}}$ - -) **where** $lCoP3^{ab}\ \iota\ \eta \equiv (\iota\ (\eta\ a)\ (\eta\ b{::}\sigma)) \preceq (\iota\ b\ a)$
definition $lCoPw\ \ \iota\ \eta \equiv \forall\, a\ b.\ lCoPw^{ab}\ \iota\ \eta$
definition $lCoP1\ \ \iota\ \eta \equiv \forall\, a\ b.\ lCoP1^{ab}\ \iota\ \eta$
definition $lCoP1'\ \iota\ \eta \equiv \forall\, a\ b.\ (\iota\ a\ (\eta\ b)) \approx (\iota\ b\ (\eta\ a))$
definition $lCoP2\ \ \iota\ \eta \equiv \forall\, a\ b.\ lCoP2^{ab}\ \iota\ \eta$
definition $lCoP2'\ \iota\ \eta \equiv \forall\, a\ b.\ (\iota\ (\eta\ a)\ b) \approx (\iota\ (\eta\ b)\ a)$
definition $lCoP3\ \ \iota\ \eta \equiv \forall\, a\ b.\ lCoP3^{ab}\ \iota\ \eta$
declare *lCoPw-def*[*Defs*] *lCoP1-def*[*Defs*] *lCoP1'-def*[*Defs*]
lCoP2-def[*Defs*] *lCoP2'-def*[*Defs*] *lCoP3-def*[*Defs*]

lemma *lCoP1-defs-rel*: $lCoP1\ \iota\ \eta = lCoP1'\ \iota\ \eta$ ⟨*proof*⟩
lemma *lCoP2-defs-rel*: $lCoP2\ \iota\ \eta = lCoP2'\ \iota\ \eta$ ⟨*proof*⟩

All local contraposition variants are in general independent from each other. However if we take classical implication we can verify some relationships.

lemma *lCoP1-def2*: $lCoP1(\rightarrow)\ \eta = (lCoPw(\rightarrow)\ \eta \wedge DNI\ \eta)$ ⟨*proof*⟩
lemma *lCoP2-def2*: $lCoP2(\rightarrow)\ \eta = (lCoPw(\rightarrow)\ \eta \wedge DNE\ \eta)$ ⟨*proof*⟩

lemma $lCoP1(\rightarrow)\ \eta \Longrightarrow lCoPw(\rightarrow)\ \eta$ ⟨*proof*⟩
lemma $lCoPw(\rightarrow)\ \eta \Longrightarrow lCoP1(\rightarrow)\ \eta$ **nitpick** ⟨*proof*⟩
lemma $lCoP2(\rightarrow)\ \eta \Longrightarrow lCoPw(\rightarrow)\ \eta$ ⟨*proof*⟩
lemma $lCoPw(\rightarrow)\ \eta \Longrightarrow lCoP2(\rightarrow)\ \eta$ **nitpick** ⟨*proof*⟩
lemma $lCoP3(\rightarrow)\ \eta \Longrightarrow lCoPw(\rightarrow)\ \eta$ ⟨*proof*⟩
lemma $lCoPw(\rightarrow)\ \eta \Longrightarrow lCoP3(\rightarrow)\ \eta$ **nitpick** ⟨*proof*⟩

lemma *lCoP123*: $lCoP1(\rightarrow)\ \eta \wedge lCoP2(\rightarrow)\ \eta \Longrightarrow lCoP3(\rightarrow)\ \eta$ ⟨*proof*⟩

Local variants imply global ones as expected.

lemma $lCoPw(\rightarrow)\ \eta \Longrightarrow CoPw\ \eta$ ⟨*proof*⟩
lemma $lCoP1(\rightarrow)\ \eta \Longrightarrow CoP1\ \eta$ ⟨*proof*⟩
lemma $lCoP2(\rightarrow)\ \eta \Longrightarrow CoP2\ \eta$ ⟨*proof*⟩
lemma $lCoP3(\rightarrow)\ \eta \Longrightarrow CoP3\ \eta$ ⟨*proof*⟩

Explore some (non)entailment relations.

lemma *lCoPw-XCoP*: $lCoPw(\rightarrow)\ \eta = XCoP\ \eta$ ⟨*proof*⟩
lemma *lCoP1-TND*: $lCoP1(\rightarrow)\ \eta \Longrightarrow TND\ \eta$ ⟨*proof*⟩
lemma $TND\ \eta \Longrightarrow lCoP1(\rightarrow)\ \eta$ **nitpick** ⟨*proof*⟩
lemma *lCoP2-ECQ*: $lCoP2(\rightarrow)\ \eta \Longrightarrow ECQ\ \eta$ ⟨*proof*⟩
lemma $ECQ\ \eta \Longrightarrow lCoP2(\rightarrow)\ \eta$ **nitpick** ⟨*proof*⟩

3.1.9 Local modus tollens axioms

lMT: Modus tollens (local/axiom variants).

abbreviation *plMT0* ($lMT0^{--}$ - -) **where** $lMT0^{ab}\ \iota\ \eta \equiv (\iota\ a\ b{::}\sigma) \wedge (\eta\ b) \preceq (\eta\ a)$
abbreviation *plMT1* ($lMT1^{--}$ - -) **where** $lMT1^{ab}\ \iota\ \eta \equiv (\iota\ a\ (\eta\ b{::}\sigma)) \wedge b \preceq (\eta\ a)$
abbreviation *plMT2* ($lMT2^{--}$ - -) **where** $lMT2^{ab}\ \iota\ \eta \equiv (\iota\ (\eta\ a)\ b{::}\sigma) \wedge (\eta\ b) \preceq a$
abbreviation *plMT3* ($lMT3^{--}$ - -) **where** $lMT3^{ab}\ \iota\ \eta \equiv (\iota\ (\eta\ a)\ (\eta\ b{::}\sigma)) \wedge b \preceq a$
definition $lMT0\ \iota\ \eta \equiv \forall\, a\ b.\ lMT0^{ab}\ \iota\ \eta$
definition $lMT1\ \iota\ \eta \equiv \forall\, a\ b.\ lMT1^{ab}\ \iota\ \eta$
definition $lMT2\ \iota\ \eta \equiv \forall\, a\ b.\ lMT2^{ab}\ \iota\ \eta$
definition $lMT3\ \iota\ \eta \equiv \forall\, a\ b.\ lMT3^{ab}\ \iota\ \eta$
declare *lMT0-def*[*Defs*] *lMT1-def*[*Defs*] *lMT2-def*[*Defs*] *lMT3-def*[*Defs*]

All local MT variants are in general independent from each other and also from local CoP instances. However if we take classical implication we can verify that local MT and CoP are indeed equivalent.

lemma $lMT0(\rightarrow)\ \eta = lCoPw(\rightarrow)\ \eta$ ⟨*proof*⟩
lemma $lMT1(\rightarrow)\ \eta = lCoP1(\rightarrow)\ \eta$ ⟨*proof*⟩
lemma $lMT2(\rightarrow)\ \eta = lCoP2(\rightarrow)\ \eta$ ⟨*proof*⟩
lemma $lMT3(\rightarrow)\ \eta = lCoP3(\rightarrow)\ \eta$ ⟨*proof*⟩

3.1.10 Disjunctive syllogism

DS: disjunctive syllogism.

abbreviation *pDS1* ($DS1^{--}$ - -) **where** $DS1^{ab}\ \iota\ \eta \equiv (a \vee b{::}\sigma) \preceq (\iota\ (\eta\ a)\ b)$
abbreviation *pDS2* ($DS2^{--}$ - -) **where** $DS2^{ab}\ \iota\ \eta \equiv (\iota\ (\eta\ a)\ b{::}\sigma) \preceq (a \vee b)$
abbreviation *pDS3* ($DS3^{--}$ - -) **where** $DS3^{ab}\ \iota\ \eta \equiv ((\eta\ a) \vee b{::}\sigma) \preceq (\iota\ a\ b)$
abbreviation *pDS4* ($DS4^{--}$ - -) **where** $DS4^{ab}\ \iota\ \eta \equiv (\iota\ a\ b{::}\sigma) \preceq ((\eta\ a) \vee b)$
definition $DS1\ \iota\ \eta \equiv \forall\, a\ b.\ DS1^{ab}\ \iota\ \eta$
definition $DS2\ \iota\ \eta \equiv \forall\, a\ b.\ DS2^{ab}\ \iota\ \eta$
definition $DS3\ \iota\ \eta \equiv \forall\, a\ b.\ DS3^{ab}\ \iota\ \eta$
definition $DS4\ \iota\ \eta \equiv \forall\, a\ b.\ DS4^{ab}\ \iota\ \eta$
declare *DS1-def*[*Defs*] *DS2-def*[*Defs*] *DS3-def*[*Defs*] *DS4-def*[*Defs*]

All DS variants are in general independent from each other. However if we take classical implication we can verify that the pairs DS1-DS3 and DS2-DS4 are indeed equivalent.

lemma $DS1(\rightarrow)\ \eta = DS3(\rightarrow)\ \eta$ ⟨*proof*⟩
lemma $DS2(\rightarrow)\ \eta = DS4(\rightarrow)\ \eta$ ⟨*proof*⟩

Explore some (non)entailment relations.

lemma *DS1-nDNor*: *DS1*(→) $\eta \Longrightarrow$ *nDNor* η ⟨*proof*⟩
lemma *DS2-nNor*: *DS2*(→) $\eta \Longrightarrow$ *nNor* η ⟨*proof*⟩
lemma *lCoP2-DS1*: *lCoP2*(→) $\eta \Longrightarrow$ *DS1*(→) η ⟨*proof*⟩
lemma *lCoP1-DS2*: *lCoP1*(→) $\eta \Longrightarrow$ *DS2*(→) η ⟨*proof*⟩
lemma *CoP2* $\eta \Longrightarrow$ *DS1*(→) η **nitpick** ⟨*proof*⟩
lemma *CoP1* $\eta \Longrightarrow$ *DS2*(→) η **nitpick** ⟨*proof*⟩

end
theory *sse-operation-negative-quantification*
imports *sse-operation-negative sse-boolean-algebra-quantification*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

3.2 Definitions and interrelations (infinitary case)

We define and interrelate infinitary variants for some previously introduced ('negative') conditions on operations. We show how they relate to quantifiers as previously defined.

iDM: infinitary De Morgan laws.

abbreviation *riDM1* (*iDM1*$^{-}$ -) **where** $iDM1^S\ \eta \equiv \eta(\bigvee S) \preceq \bigwedge Ra[\eta|S]$
abbreviation *riDM2* (*iDM2*$^{-}$ -) **where** $iDM2^S\ \eta \equiv \bigvee Ra[\eta|S] \preceq \eta(\bigwedge S)$
abbreviation *riDM3* (*iDM3*$^{-}$ -) **where** $iDM3^S\ \eta \equiv \eta(\bigwedge S) \preceq \bigvee Ra[\eta|S]$
abbreviation *riDM4* (*iDM4*$^{-}$ -) **where** $iDM4^S\ \eta \equiv \bigwedge Ra[\eta|S] \preceq \eta(\bigvee S)$
definition $iDM1\ \eta \equiv \forall S.\ iDM1^S\ \eta$
definition $iDM2\ \eta \equiv \forall S.\ iDM2^S\ \eta$
definition $iDM3\ \eta \equiv \forall S.\ iDM3^S\ \eta$
definition $iDM4\ \eta \equiv \forall S.\ iDM4^S\ \eta$
declare *iDM1-def*[*Defs*] *iDM2-def*[*Defs*] *iDM3-def*[*Defs*] *iDM4-def*[*Defs*]

lemma *CoPw-iDM1*: *CoPw* $\eta \Longrightarrow$ *iDM1* η ⟨*proof*⟩
lemma *CoPw-iDM2*: *CoPw* $\eta \Longrightarrow$ *iDM2* η ⟨*proof*⟩
lemma *CoP2-iDM3*: *CoP2* $\eta \Longrightarrow$ *iDM3* η ⟨*proof*⟩
lemma *CoP1-iDM4*: *CoP1* $\eta \Longrightarrow$ *iDM4* η ⟨*proof*⟩
lemma *XCoP* $\eta \Longrightarrow$ *iDM3* η **nitpick** ⟨*proof*⟩
lemma *XCoP* $\eta \Longrightarrow$ *iDM4* η **nitpick** ⟨*proof*⟩

DM1, DM2, iDM1, iDM2 and CoPw are equivalent.

lemma *iDM1-rel*: *iDM1* $\eta \Longrightarrow$ *DM1* η ⟨*proof*⟩
lemma *iDM2-rel*: *iDM2* $\eta \Longrightarrow$ *DM2* η ⟨*proof*⟩
lemma *DM1* $\eta =$ *iDM1* η ⟨*proof*⟩
lemma *DM2* $\eta =$ *iDM2* η ⟨*proof*⟩
lemma *iDM1* $\eta =$ *iDM2* η ⟨*proof*⟩

iDM3/4 entail their finitary variants but not the other way round.

lemma *iDM3-rel*: *iDM3* $\eta \Longrightarrow$ *DM3* η ⟨*proof*⟩
lemma *iDM4-rel*: *iDM4* $\eta \Longrightarrow$ *DM4* η ⟨*proof*⟩
lemma *DM3* $\eta \Longrightarrow$ *iDM3* η **nitpick** ⟨*proof*⟩
lemma *DM4* $\eta \Longrightarrow$ *iDM4* η **nitpick** ⟨*proof*⟩

Indeed the previous characterization of the infinitary De Morgan laws is fairly general and entails the traditional version employing quantifiers (though not the other way round).

The first two variants DM1/2 follow easily from DM1/2, iDM1/2 or CoPw (all of them equivalent).

lemma *iDM1-trad*: *iDM1* $\eta \Longrightarrow \forall \pi.\ \eta(\exists x.\ \pi\ x) \preceq (\forall x.\ \eta(\pi\ x))$ ⟨*proof*⟩
lemma *iDM2-trad*: *iDM2* $\eta \Longrightarrow \forall \pi.\ (\exists x.\ \eta(\pi\ x)) \preceq \eta(\forall x.\ \pi\ x)$ ⟨*proof*⟩

An analogous relationship holds for variants DM3/4, though the proof is less trivial. To see how let us first consider an intermediate version of the De Morgan laws, obtained as a particular case of the general variant above, with S as the range of a propositional function.

abbreviation *piDM1* $\pi\ \eta \equiv \eta(\bigvee Ra(\pi)) \preceq \bigwedge Ra[\eta|Ra(\pi)]$
abbreviation *piDM2* $\pi\ \eta \equiv \bigvee Ra[\eta|Ra(\pi)] \preceq \eta(\bigwedge Ra(\pi))$
abbreviation *piDM3* $\pi\ \eta \equiv \eta(\bigwedge Ra(\pi)) \preceq \bigvee Ra[\eta|Ra(\pi)]$
abbreviation *piDM4* $\pi\ \eta \equiv \bigwedge Ra[\eta|Ra(\pi)] \preceq \eta(\bigvee Ra(\pi))$

They are entailed (unidirectionally) by the general De Morgan laws.

lemma *iDM1* $\eta \Longrightarrow \forall \pi.$ *piDM1* $\pi\ \eta$ ⟨*proof*⟩
lemma *iDM2* $\eta \Longrightarrow \forall \pi.$ *piDM2* $\pi\ \eta$ ⟨*proof*⟩
lemma *iDM3* $\eta \Longrightarrow \forall \pi.$ *piDM3* $\pi\ \eta$ ⟨*proof*⟩
lemma *iDM4* $\eta \Longrightarrow \forall \pi.$ *piDM4* $\pi\ \eta$ ⟨*proof*⟩

Drawing upon the relationships shown previously we can rewrite the latter two as:

lemma *iDM3-aux*: *piDM3* $\pi\ \eta \equiv \eta(\forall \pi) \preceq \exists\,[\eta|(\!|Ra\ \pi|\!)]^N$ ⟨*proof*⟩
lemma *iDM4-aux*: *piDM4* $\pi\ \eta \equiv \forall\,[\eta|(\!|Ra\ \pi|\!)]^P \preceq \eta(\exists \pi)$ ⟨*proof*⟩

and thus finally obtain the desired formulas.

lemma *iDM3-trad*: *iDM3* $\eta \Longrightarrow \forall \pi.\ \eta(\forall x.\ \pi\ x) \preceq (\exists x.\ \eta(\pi\ x))$ ⟨*proof*⟩
lemma *iDM4-trad*: *iDM4* $\eta \Longrightarrow \forall \pi.\ (\forall x.\ \eta(\pi\ x)) \preceq \eta(\exists x.\ \pi\ x)$ ⟨*proof*⟩

end
theory *topo-operators-basic*
imports *sse-operation-positive-quantification*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

abbreviation *implies-rl*::*bool*⇒*bool*⇒*bool* (**infixl** ⟵ *25*) **where** $\varphi \longleftarrow \psi \equiv \psi \longrightarrow \varphi$

4 Topological operators

Below we define some conditions on algebraic operations (aka. operators) with type $\sigma \Rightarrow \sigma$. Those operations are aimed at extending a Boolean 'algebra of propositions' towards different generalizations of topological algebras. We divide this section into two parts. In the first we define and interrelate the topological operators of interior, closure, border and frontier. In the second we introduce the (more fundamental) notion of derivative (aka. derived set) and its related notion of (Cantorian) coherence, defining both as operators. We follow the naming conventions introduced originally by Kuratowski [8] (cf. also [9]) and Zarycki [12].

4.1 Interior and closure

In this section we examine the traditional notion of topological (closure, resp. interior) algebras in the spirit of McKinsey & Tarski [11], but drawing primarily from the works of Zarycki [12] and Kuratowski [8]. We also explore the less-known notions of border (cf. 'Rand' [6], 'bord' [12]) and frontier (aka. 'boundary'; cf. 'Grenze' [6], 'frontière' [12] [9]) as studied by Zarycki [12] and define corresponding operations for them.

4.1.1 Interior conditions

abbreviation *Int-1* $\varphi \equiv$ *MULT* φ
abbreviation *Int-1a* $\varphi \equiv$ *MULT-a* φ
abbreviation *Int-1b* $\varphi \equiv$ *MULT-b* φ
abbreviation *Int-2* $\varphi \equiv$ *dEXP* φ
abbreviation *Int-3* $\varphi \equiv$ *dNOR* φ
abbreviation *Int-4* $\varphi \equiv$ *IDEM* φ
abbreviation *Int-4'* $\varphi \equiv$ *IDEMa* φ

abbreviation *Int-5* $\varphi \equiv$ *NOR* φ
definition *Int-6* $\varphi \equiv \forall A\ B.\ \varphi(A \leftharpoonup B) \preceq \varphi(A) \leftharpoonup \varphi(B)$
definition *Int-7* $\varphi \equiv \forall A\ B.\ \varphi(A \rightarrow B) \preceq \varphi(A) \rightarrow \varphi(B)$
definition *Int-8* $\varphi \equiv \forall A\ B.\ \varphi(\varphi\ A \vee \varphi\ B) \approx (\varphi\ A) \vee (\varphi\ B)$
definition *Int-9* $\varphi \equiv \forall A\ B.\ \varphi\ A \preceq B \longrightarrow \varphi\ A \preceq \varphi\ B$

φ is an interior operator ($\mathfrak{I}(\varphi)$) iff it satisfies conditions 1-4 (cf. [12] and also [9]). This characterization is shown consistent by generating a non-trivial model.

abbreviation $\mathfrak{I}\ \varphi \equiv$ *Int-1* $\varphi \wedge$ *Int-2* $\varphi \wedge$ *Int-3* $\varphi \wedge$ *Int-4* φ
lemma $\mathfrak{I}\ \varphi$ **nitpick**[*satisfy*, *card w=3*] ⟨*proof*⟩

We verify some properties which will become useful later (also to improve provers' performance).

lemma *PI1*: *Int-1* $\varphi =$ (*Int-1a* $\varphi \wedge$ *Int-1b* φ) ⟨*proof*⟩
lemma *PI4*: *Int-2* $\varphi \Longrightarrow$ (*Int-4* $\varphi =$ *Int-4'* φ) ⟨*proof*⟩
lemma *PI5*: *Int-2* $\varphi \Longrightarrow$ *Int-5* φ ⟨*proof*⟩
lemma *PI6*: *Int-1a* $\varphi \Longrightarrow$ *Int-2* $\varphi \Longrightarrow$ *Int-6* φ ⟨*proof*⟩
lemma *PI7*: *Int-1* $\varphi \Longrightarrow$ *Int-7* φ ⟨*proof*⟩
lemma *PI8*: *Int-1a* $\varphi \Longrightarrow$ *Int-2* $\varphi \Longrightarrow$ *Int-4* $\varphi \Longrightarrow$ *Int-8* φ ⟨*proof*⟩
lemma *PI9*: *Int-1a* $\varphi \Longrightarrow$ *Int-4* $\varphi \Longrightarrow$ *Int-9* φ ⟨*proof*⟩

4.1.2 Closure conditions

abbreviation *Cl-1* $\varphi \equiv$ *ADDI* φ
abbreviation *Cl-1a* $\varphi \equiv$ *ADDI-a* φ
abbreviation *Cl-1b* $\varphi \equiv$ *ADDI-b* φ
abbreviation *Cl-2* $\varphi \equiv$ *EXP* φ
abbreviation *Cl-3* $\varphi \equiv$ *NOR* φ
abbreviation *Cl-4* $\varphi \equiv$ *IDEM* φ
abbreviation *Cl-4'* $\varphi \equiv$ *IDEMb* φ

abbreviation *Cl-5* $\varphi \equiv$ *dNOR* φ
definition *Cl-6* $\varphi \equiv \forall A\ B.\ (\varphi\ A) \leftharpoonup (\varphi\ B) \preceq \varphi\ (A \leftharpoonup B)$
definition *Cl-7* $\varphi \equiv \forall A\ B.\ (\varphi\ A) \rightarrow (\varphi\ B) \preceq \varphi\ (A \rightarrow B)$
definition *Cl-8* $\varphi \equiv \forall A\ B.\ \varphi(\varphi\ A \wedge \varphi\ B) \approx (\varphi\ A) \wedge (\varphi\ B)$
definition *Cl-9* $\varphi \equiv \forall A\ B.\ A \preceq \varphi\ B \longrightarrow \varphi\ A \preceq \varphi\ B$

φ is a closure operator ($\mathfrak{C}(\varphi)$) iff it satisfies conditions 1-4 (cf. [8] [9]). This characterization is shown consistent by generating a non-trivial model.

abbreviation $\mathfrak{C}\ \varphi \equiv$ *Cl-1* $\varphi \wedge$ *Cl-2* $\varphi \wedge$ *Cl-3* $\varphi \wedge$ *Cl-4* φ
lemma $\mathfrak{C}\ \varphi$ **nitpick**[*satisfy*, *card w=3*] ⟨*proof*⟩

We verify some properties that will become useful later.

lemma *PC1*: *Cl-1* $\varphi =$ (*Cl-1a* $\varphi \wedge$ *Cl-1b* φ) ⟨*proof*⟩
lemma *PC4*: *Cl-2* $\varphi \Longrightarrow$ (*Cl-4* $\varphi =$ *Cl-4'* φ) ⟨*proof*⟩
lemma *PC5*: *Cl-2* $\varphi \Longrightarrow$ *Cl-5* φ ⟨*proof*⟩

lemma *PC6*: $\textit{Cl-1}\ \varphi \Longrightarrow \textit{Cl-6}\ \varphi$ ⟨*proof*⟩
lemma *PC7*: $\textit{Cl-1b}\ \varphi \Longrightarrow \textit{Cl-2}\ \varphi \Longrightarrow \textit{Cl-7}\ \varphi$ ⟨*proof*⟩
lemma *PC8*: $\textit{Cl-1b}\ \varphi \Longrightarrow \textit{Cl-2}\ \varphi \Longrightarrow \textit{Cl-4}\ \varphi \Longrightarrow \textit{Cl-8}\ \varphi$ ⟨*proof*⟩
lemma *PC9*: $\textit{Cl-1b}\ \varphi \Longrightarrow \textit{Cl-4}\ \varphi \Longrightarrow \textit{Cl-9}\ \varphi$ ⟨*proof*⟩

4.1.3 Exploring dualities

lemma *IC1-dual*: $\textit{Int-1a}\ \varphi = \textit{Cl-1b}\ \varphi$ ⟨*proof*⟩
lemma $\textit{Int-1b}\ \varphi = \textit{Cl-1a}\ \varphi$ **nitpick** ⟨*proof*⟩

lemma *IC1a*: $\textit{Int-1a}\ \varphi \Longrightarrow \textit{Cl-1b}\ \varphi^d$ ⟨*proof*⟩
lemma *IC1b*: $\textit{Int-1b}\ \varphi \Longrightarrow \textit{Cl-1a}\ \varphi^d$ ⟨*proof*⟩
lemma *IC1*: $\textit{Int-1}\ \varphi \Longrightarrow \textit{Cl-1}\ \varphi^d$ ⟨*proof*⟩
lemma *IC2*: $\textit{Int-2}\ \varphi \Longrightarrow \textit{Cl-2}\ \varphi^d$ ⟨*proof*⟩
lemma *IC3*: $\textit{Int-3}\ \varphi \Longrightarrow \textit{Cl-3}\ \varphi^d$ ⟨*proof*⟩
lemma *IC4*: $\textit{Int-4}\ \varphi \Longrightarrow \textit{Cl-4}\ \varphi^d$ ⟨*proof*⟩
lemma *IC4'*: $\textit{Int-4'}\ \varphi \Longrightarrow \textit{Cl-4'}\ \varphi^d$ ⟨*proof*⟩
lemma *IC5*: $\textit{Int-5}\ \varphi \Longrightarrow \textit{Cl-5}\ \varphi^d$ ⟨*proof*⟩

lemma *CI1a*: $\textit{Cl-1a}\ \varphi \Longrightarrow \textit{Int-1b}\ \varphi^d$ ⟨*proof*⟩
lemma *CI1b*: $\textit{Cl-1b}\ \varphi \Longrightarrow \textit{Int-1a}\ \varphi^d$ ⟨*proof*⟩
lemma *CI1*: $\textit{Cl-1}\ \varphi \Longrightarrow \textit{Int-1}\ \varphi^d$ ⟨*proof*⟩
lemma *CI2*: $\textit{Cl-2}\ \varphi \Longrightarrow \textit{Int-2}\ \varphi^d$ ⟨*proof*⟩
lemma *CI3*: $\textit{Cl-3}\ \varphi \Longrightarrow \textit{Int-3}\ \varphi^d$ ⟨*proof*⟩
lemma *CI4*: $\textit{Cl-4}\ \varphi \Longrightarrow \textit{Int-4}\ \varphi^d$ ⟨*proof*⟩
lemma *CI4'*: $\textit{Cl-4'}\ \varphi \Longrightarrow \textit{Int-4'}\ \varphi^d$ ⟨*proof*⟩
lemma *CI5*: $\textit{Cl-5}\ \varphi \Longrightarrow \textit{Int-5}\ \varphi^d$ ⟨*proof*⟩

4.2 Frontier and border

4.2.1 Frontier conditions

definition $\textit{Fr-1a}\ \varphi \equiv \forall A\ B.\ (A \wedge B) \wedge \varphi(A \wedge B) \preceq (A \wedge B) \wedge (\varphi\ A \vee \varphi\ B)$
definition $\textit{Fr-1b}\ \varphi \equiv \forall A\ B.\ (A \wedge B) \wedge \varphi(A \wedge B) \succeq (A \wedge B) \wedge (\varphi\ A \vee \varphi\ B)$
definition $\textit{Fr-1}\ \varphi \equiv \forall A\ B.\ (A \wedge B) \wedge \varphi(A \wedge B) \approx (A \wedge B) \wedge (\varphi\ A \vee \varphi\ B)$
definition $\textit{Fr-2}\ \varphi \equiv \forall A.\ \varphi\ A \approx \varphi(-A)$
abbreviation $\textit{Fr-3}\ \varphi \equiv \textit{NOR}\ \varphi$
definition $\textit{Fr-4}\ \varphi \equiv \forall A.\ \varphi(\varphi\ A) \preceq (\varphi\ A)$

definition $\textit{Fr-5}\ \varphi \equiv \forall A.\ \varphi(\varphi(\varphi\ A)) \approx \varphi(\varphi\ A)$
definition $\textit{Fr-6}\ \varphi \equiv \forall A\ B.\ A \preceq B \longrightarrow (\varphi\ A \preceq B \vee \varphi\ B)$

φ is a topological frontier operator ($\mathfrak{F}(\varphi)$) iff it satisfies conditions 1-4 (cf. [12]). This is also shown consistent by generating a non-trivial model.

abbreviation $\mathfrak{F}\ \varphi \equiv \textit{Fr-1}\ \varphi \wedge \textit{Fr-2}\ \varphi \wedge \textit{Fr-3}\ \varphi \wedge \textit{Fr-4}\ \varphi$
lemma $\mathfrak{F}\ \varphi$ **nitpick**[*satisfy*, *card w=3*] ⟨*proof*⟩

We now verify some useful properties of the frontier operator.

lemma *PF1*: $\textit{Fr-1}\ \varphi = (\textit{Fr-1a}\ \varphi \wedge \textit{Fr-1b}\ \varphi)$ ⟨*proof*⟩
lemma *PF5*: $\textit{Fr-1}\ \varphi \Longrightarrow \textit{Fr-4}\ \varphi \Longrightarrow \textit{Fr-5}\ \varphi$ ⟨*proof*⟩
lemma *PF6*: $\textit{Fr-1b}\ \varphi \Longrightarrow \textit{Fr-2}\ \varphi \Longrightarrow \textit{Fr-6}\ \varphi$ ⟨*proof*⟩

4.2.2 Border conditions

definition $\textit{Br-1}\ \varphi \equiv \forall A\ B.\ \varphi(A \wedge B) \approx (A \wedge \varphi\ B) \vee (B \wedge \varphi\ A)$
definition $\textit{Br-2}\ \varphi \equiv (\varphi\ \top) \approx \bot$

definition *Br-3* $\varphi \equiv \forall A.\ \varphi(\varphi^d\ A) \preceq A$

definition *Br-4* $\varphi \equiv \forall A\ B.\ A \preceq B \longrightarrow A \wedge (\varphi\ B) \preceq \varphi\ A$
definition *Br-5a* $\varphi \equiv \forall A.\ \varphi(\varphi^d\ A) \preceq \varphi\ A$
definition *Br-5b* $\varphi \equiv \forall A.\ \varphi\ A \preceq A$
definition *Br-5c* $\varphi \equiv \forall A.\ A \preceq \varphi^d\ A$
definition *Br-5d* $\varphi \equiv \forall A.\ \varphi^d\ A \preceq \varphi^d(\varphi\ A)$
abbreviation *Br-6* $\varphi \equiv$ *IDEM* φ
abbreviation *Br-7* $\varphi \equiv$ *ADDI-a* φ
abbreviation *Br-8* $\varphi \equiv$ *MULT-b* φ
definition *Br-9* $\varphi \equiv \forall A\ B.\ \varphi(A \wedge B) \preceq (\varphi\ A) \vee (\varphi\ B)$
definition *Br-10* $\varphi \equiv \forall A.\ \varphi(-(\varphi\ A) \wedge \varphi^d\ A) \approx \bot$

φ is a topological border operator ($\mathfrak{B}(\varphi)$) iff it satisfies conditions 1-3 (cf. [12]). This is also shown consistent.

abbreviation $\mathfrak{B}\ \varphi\ \equiv$ *Br-1* $\varphi \wedge$ *Br-2* $\varphi \wedge$ *Br-3* φ
lemma $\mathfrak{B}\ \varphi$ **nitpick**[*satisfy*, *card w=3*] ⟨*proof*⟩

We now verify some useful properties of the border operator.

lemma *PB4*: *Br-1* $\varphi \Longrightarrow$ *Br-4* φ ⟨*proof*⟩
lemma *PB5b*: *Br-1* $\varphi \Longrightarrow$ *Br-5b* φ ⟨*proof*⟩
lemma *PB5c*: *Br-1* $\varphi \Longrightarrow$ *Br-5c* φ ⟨*proof*⟩
lemma *PB5a*: *Br-1* $\varphi \Longrightarrow$ *Br-3* $\varphi \Longrightarrow$ *Br-5a* φ ⟨*proof*⟩
lemma *PB5d*: *Br-1* $\varphi \Longrightarrow$ *Br-3* $\varphi \Longrightarrow$ *Br-5d* φ ⟨*proof*⟩
lemma *PB6*: *Br-1* $\varphi \Longrightarrow$ *Br-6* φ ⟨*proof*⟩
lemma *PB7*: *Br-1* $\varphi \Longrightarrow$ *Br-7* φ ⟨*proof*⟩
lemma *PB8*: *Br-1* $\varphi \Longrightarrow$ *Br-8* φ ⟨*proof*⟩
lemma *PB9*: *Br-1* $\varphi \Longrightarrow$ *Br-9* φ ⟨*proof*⟩
lemma *PB10*: *Br-1* $\varphi \Longrightarrow$ *Br-3* $\varphi \Longrightarrow$ *Br-10* φ ⟨*proof*⟩

4.2.3 Relation with closure and interior

We define and verify some conversion operators useful to derive border and frontier operators from closure/interior operators and also between each other.

Frontier operator as derived from interior.

definition *Fr-int*::$(\sigma{\Rightarrow}\sigma){\Rightarrow}(\sigma{\Rightarrow}\sigma)$ $(\mathcal{F}_I)$ **where** $\mathcal{F}_I\ \mathcal{I} \equiv \lambda A.\ -(\mathcal{I}\ A) \wedge \mathcal{I}^d\ A$
lemma *FI1*: *Int-1* $\varphi \Longrightarrow$ *Int-2* $\varphi \Longrightarrow$ *Fr-1*$(\mathcal{F}_I\ \varphi)$ ⟨*proof*⟩
lemma *FI2*: *Fr-2*$(\mathcal{F}_I\ \varphi)$ ⟨*proof*⟩
lemma *FI3*: *Int-3* $\varphi \Longrightarrow$ *Fr-3*$(\mathcal{F}_I\ \varphi)$ ⟨*proof*⟩
lemma *FI4*: *Int-1a* $\varphi \Longrightarrow$ *Int-2* $\varphi \Longrightarrow$ *Int-4* $\varphi \Longrightarrow$ *Fr-4*$(\mathcal{F}_I\ \varphi)$ ⟨*proof*⟩

Frontier operator as derived from closure.

definition *Fr-cl*::$(\sigma{\Rightarrow}\sigma){\Rightarrow}(\sigma{\Rightarrow}\sigma)$ $(\mathcal{F}_C)$ **where** $\mathcal{F}_C\ \mathcal{C} \equiv \lambda A.\ (\mathcal{C}\ A) \wedge \mathcal{C}(-A)$
lemma *FC1*: *Cl-1* $\varphi \Longrightarrow$ *Cl-2* $\varphi \Longrightarrow$ *Fr-1*$(\mathcal{F}_C\ \varphi)$ ⟨*proof*⟩
lemma *FC2*: *Fr-2*$(\mathcal{F}_C\ \varphi)$ ⟨*proof*⟩
lemma *FC3*: *Cl-3* $\varphi \Longrightarrow$ *Fr-3*$(\mathcal{F}_C\ \varphi)$ ⟨*proof*⟩
lemma *FC4*: *Cl-1b* $\varphi \Longrightarrow$ *Cl-2* $\varphi \Longrightarrow$ *Cl-4* $\varphi \Longrightarrow$ *Fr-4*$(\mathcal{F}_C\ \varphi)$ ⟨*proof*⟩

Frontier operator as derived from border.

definition *Fr-br*::$(\sigma{\Rightarrow}\sigma){\Rightarrow}(\sigma{\Rightarrow}\sigma)$ $(\mathcal{F}_B)$ **where** $\mathcal{F}_B\ \mathcal{B} \equiv \lambda A.\ \mathcal{B}\ A \vee \mathcal{B}(-A)$
lemma *FB1*: *Br-1* $\varphi \Longrightarrow$ *Fr-1*$(\mathcal{F}_B\ \varphi)$ ⟨*proof*⟩
lemma *FB2*: *Fr-2*$(\mathcal{F}_B\ \varphi)$ ⟨*proof*⟩
lemma *FB3*: *Br-1* $\varphi \Longrightarrow$ *Br-2* $\varphi \Longrightarrow$ *Fr-3*$(\mathcal{F}_B\ \varphi)$ ⟨*proof*⟩

lemma *FB4*: *Br-1* $\varphi \Longrightarrow$ *Br-3* $\varphi \Longrightarrow$ *Fr-4*$(\mathcal{F}_B\ \varphi)$ $\langle proof \rangle$

Border operator as derived from interior.

definition *Br-int*::$(\sigma\Rightarrow\sigma)\Rightarrow(\sigma\Rightarrow\sigma)$ $(\mathcal{B}_I)$ **where** $\mathcal{B}_I\ \mathcal{I} \equiv \lambda A.\ A \leftharpoonup (\mathcal{I}\ A)$
lemma *BI1*: *Int-1* $\varphi \Longrightarrow$ *Br-1* $(\mathcal{B}_I\ \varphi)$ $\langle proof \rangle$
lemma *BI2*: *Int-3* $\varphi \Longrightarrow$ *Br-2* $(\mathcal{B}_I\ \varphi)$ $\langle proof \rangle$
lemma *BI3*: *Int-1a* $\varphi \Longrightarrow$ *Int-4* $\varphi \Longrightarrow$ *Br-3* $(\mathcal{B}_I\ \varphi)$ $\langle proof \rangle$

Border operator as derived from closure.

definition *Br-cl*::$(\sigma\Rightarrow\sigma)\Rightarrow(\sigma\Rightarrow\sigma)$ $(\mathcal{B}_C)$ **where** $\mathcal{B}_C\ \mathcal{C} \equiv \lambda A.\ A \wedge \mathcal{C}(-A)$
lemma *BC1*: *Cl-1* $\varphi \Longrightarrow$ *Br-1*$(\mathcal{B}_C\ \varphi)$ $\langle proof \rangle$
lemma *BC2*: *Cl-3* $\varphi \Longrightarrow$ *Br-2*$(\mathcal{B}_C\ \varphi)$ $\langle proof \rangle$
lemma *BC3*: *Cl-1b* $\varphi \Longrightarrow$ *Cl-4* $\varphi \Longrightarrow$ *Br-3* $(\mathcal{B}_C\ \varphi)$ $\langle proof \rangle$

Note that the previous two conversion functions are related:

lemma *BI-BC-rel*: $(\mathcal{B}_I\ \varphi) = \mathcal{B}_C(\varphi^d)$ $\langle proof \rangle$

Border operator as derived from frontier.

definition *Br-fr*::$(\sigma\Rightarrow\sigma)\Rightarrow(\sigma\Rightarrow\sigma)$ $(\mathcal{B}_F)$ **where** $\mathcal{B}_F\ \mathcal{F} \equiv \lambda A.\ A \wedge (\mathcal{F}\ A)$
lemma *BF1*: *Fr-1* $\varphi \Longrightarrow$ *Br-1*$(\mathcal{B}_F\ \varphi)$ $\langle proof \rangle$
lemma *BF2*: *Fr-2* $\varphi \Longrightarrow$ *Fr-3* $\varphi \Longrightarrow$ *Br-2*$(\mathcal{B}_F\ \varphi)$ $\langle proof \rangle$
lemma *BF3*: *Fr-1b* $\varphi \Longrightarrow$ *Fr-2* $\varphi \Longrightarrow$ *Fr-4* $\varphi \Longrightarrow$ *Br-3*$(\mathcal{B}_F\ \varphi)$ $\langle proof \rangle$

Interior operator as derived from border.

definition *Int-br*::$(\sigma\Rightarrow\sigma)\Rightarrow(\sigma\Rightarrow\sigma)$ $(\mathcal{I}_B)$ **where** $\mathcal{I}_B\ \mathcal{B} \equiv \lambda A.\ A \leftharpoonup (\mathcal{B}\ A)$
lemma *IB1*: *Br-1* $\varphi \Longrightarrow$ *Int-1*$(\mathcal{I}_B\ \varphi)$ $\langle proof \rangle$
lemma *IB2*: *Int-2*$(\mathcal{I}_B\ \varphi)$ $\langle proof \rangle$
lemma *IB3*: *Br-2* $\varphi \Longrightarrow$ *Int-3*$(\mathcal{I}_B\ \varphi)$ $\langle proof \rangle$
lemma *IB4*: *Br-1* $\varphi \Longrightarrow$ *Br-3* $\varphi \Longrightarrow$ *Int-4*$(\mathcal{I}_B\ \varphi)$ $\langle proof \rangle$

Interior operator as derived from frontier.

definition *Int-fr*::$(\sigma\Rightarrow\sigma)\Rightarrow(\sigma\Rightarrow\sigma)$ $(\mathcal{I}_F)$ **where** $\mathcal{I}_F\ \mathcal{F} \equiv \lambda A.\ A \leftharpoonup (\mathcal{F}\ A)$
lemma *IF1a*: *Fr-1b* $\varphi \Longrightarrow$ *Int-1a*$(\mathcal{I}_F\ \varphi)$ $\langle proof \rangle$
lemma *IF1b*: *Fr-1a* $\varphi \Longrightarrow$ *Int-1b*$(\mathcal{I}_F\ \varphi)$ $\langle proof \rangle$
lemma *IF1*: *Fr-1* $\varphi \Longrightarrow$ *Int-1*$(\mathcal{I}_F\ \varphi)$ $\langle proof \rangle$
lemma *IF2*: *Int-2*$(\mathcal{I}_F\ \varphi)$ $\langle proof \rangle$
lemma *IF3*: *Fr-2* $\varphi \Longrightarrow$ *Fr-3* $\varphi \Longrightarrow$ *Int-3*$(\mathcal{I}_F\ \varphi)$ $\langle proof \rangle$
lemma *IF4*: *Fr-1a* $\varphi \Longrightarrow$ *Fr-2* $\varphi \Longrightarrow$ *Fr-4* $\varphi \Longrightarrow$ *Int-4*$(\mathcal{I}_F\ \varphi)$
$\langle proof \rangle$

Closure operator as derived from border.

definition *Cl-br*::$(\sigma\Rightarrow\sigma)\Rightarrow(\sigma\Rightarrow\sigma)$ $(\mathcal{C}_B)$ **where** $\mathcal{C}_B\ \mathcal{B} \equiv \lambda A.\ A \vee \mathcal{B}(-A)$
lemma *CB1*: *Br-1* $\varphi \Longrightarrow$ *Cl-1*$(\mathcal{C}_B\ \varphi)$ $\langle proof \rangle$
lemma *CB2*: *Cl-2*$(\mathcal{C}_B\ \varphi)$ $\langle proof \rangle$
lemma *CB3*: *Br-2* $\varphi \Longrightarrow$ *Cl-3*$(\mathcal{C}_B\ \varphi)$ $\langle proof \rangle$
lemma *CB4*: *Br-1* $\varphi \Longrightarrow$ *Br-3* $\varphi \Longrightarrow$ *Cl-4*$(\mathcal{C}_B\ \varphi)$ $\langle proof \rangle$

Closure operator as derived from frontier.

definition *Cl-fr*::$(\sigma\Rightarrow\sigma)\Rightarrow(\sigma\Rightarrow\sigma)$ $(\mathcal{C}_F)$ **where** $\mathcal{C}_F\ \mathcal{F} \equiv \lambda A.\ A \vee (\mathcal{F}\ A)$
lemma *CF1b*: *Fr-1b* $\varphi \Longrightarrow$ *Fr-2* $\varphi \Longrightarrow$ *Cl-1b*$(\mathcal{C}_F\ \varphi)$ $\langle proof \rangle$
lemma *CF1a*: *Fr-1a* $\varphi \Longrightarrow$ *Fr-2* $\varphi \Longrightarrow$ *Cl-1a*$(\mathcal{C}_F\ \varphi)$ $\langle proof \rangle$
lemma *CF1*: *Fr-1* $\varphi \Longrightarrow$ *Fr-2* $\varphi \Longrightarrow$ *Cl-1*$(\mathcal{C}_F\ \varphi)$ $\langle proof \rangle$
lemma *CF2*: *Cl-2*$(\mathcal{C}_F\ \varphi)$ $\langle proof \rangle$
lemma *CF3*: *Fr-3* $\varphi \Longrightarrow$ *Cl-3*$(\mathcal{C}_F\ \varphi)$ $\langle proof \rangle$
lemma *CF4*: *Fr-1a* $\varphi \Longrightarrow$ *Fr-2* $\varphi \Longrightarrow$ *Fr-4* $\varphi \Longrightarrow$ *Cl-4*$(\mathcal{C}_F\ \varphi)$ $\langle proof \rangle$

4.2.4 Infinitary conditions

We define the essential infinitary conditions for the closure and interior operators (entailing infinite additivity and multiplicativity resp.). Observe that the other direction is implied by monotonicity (MONO).

abbreviation *Cl-inf* $\varphi \equiv$ *iADDI-a*(φ)
abbreviation *Int-inf* $\varphi \equiv$ *iMULT-b*(φ)

There exists indeed a condition on frontier operators responsible for the infinitary conditions above:

definition *Fr-inf* $\varphi \equiv \forall S.\ \bigwedge S \wedge \varphi(\bigwedge S) \preceq \bigwedge S \wedge \bigvee Ra[\varphi|S]$

lemma *CF-inf*: *Fr-2* $\varphi \Longrightarrow$ *Fr-inf* $\varphi \Longrightarrow$ *Cl-inf*$(\mathcal{C}_F\ \varphi)$ $\langle proof \rangle$
lemma *IF-inf*: *Fr-inf* $\varphi \Longrightarrow$ *Int-inf*$(\mathcal{I}_F\ \varphi)$ $\langle proof \rangle$

This condition is indeed strong enough to entail closure of the fixed-point predicates under infimum/supremum.

lemma *fp-IF-inf-closed*: *Fr-inf* $\varphi \Longrightarrow$ *infimum-closed* (*fp* $(\mathcal{I}_F\ \varphi)$) $\langle proof \rangle$
lemma *fp-CF-sup-closed*: *Fr-inf* $\varphi \Longrightarrow$ *Fr-2* $\varphi \Longrightarrow$ *supremum-closed* (*fp* $(\mathcal{C}_F\ \varphi)$) $\langle proof \rangle$

end
theory *topo-operators-derivative*
  imports *topo-operators-basic*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

4.3 Derivative and coherence

In this section we investigate two related operators, namely the 'derivative' (or 'derived set') and the (Cantorian) 'coherence' of a set. The derivative of a set is the set of its accumulation (aka. limit) points. The coherence of a set A is the set formed by those limit points of A belonging to A. For the derivative operator we draw upon the works by Kuratowski [8] and (in more detail) by Zarycki [14]; cf. also McKinsey & Tarski [11]. For the (Cantorian) coherence operator we follow the treatment given by Zarycki in [13].

4.3.1 Derivative conditions

The derivative conditions overlap partly with Kuratowski closure conditions [8]. We try to make both notations coincide when possible.

abbreviation *Der-1* $\varphi \equiv$ *Cl-1* φ
abbreviation *Der-1a* $\varphi \equiv$ *Cl-1a* φ
abbreviation *Der-1b* $\varphi \equiv$ *Cl-1b* φ
abbreviation *Der-2* $\varphi \equiv$ *Cl-5* φ — follows from Cl-2
abbreviation *Der-3* $\varphi \equiv$ *Cl-3* φ
abbreviation *Der-4* $\varphi \equiv$ *Cl-4′* φ
definition *Der-4e* $\varphi \equiv \forall A.\ \varphi(\varphi\ A) \preceq (\varphi\ A \vee A)$
definition *Der-5* $\varphi \equiv \forall A.\ (\varphi\ A \preceq A) \wedge (A \preceq \varphi^d\ A) \longrightarrow (A \approx \bot \vee A \approx \top)$

Some remarks: Condition Der-2 basically says (when assuming other derivative axioms) that the space is dense-in-itself, i.e. that all points are accumulation points (no point is isolated) w.r.t the whole space. Der-4 is a weakened (left-to-right) variant of Cl-4. Condition Der-4e corresponds to a (weaker) condition than Der-4 and is used in more recent literature (in particular in the works of Leo Esakia [5]). When other derivative axioms are assumed, Der-5 above as used by Zarycki [14] says that the only clopen sets in the space are the top and bottom elements (empty set and universe, resp.). We verify some properties:

lemma *Der4e-rel*: $Der\text{-}4\ \varphi \Longrightarrow Der\text{-}4e\ \varphi\ \langle proof \rangle$
lemma *PD1*: $Der\text{-}1b\ \varphi \Longrightarrow \forall A\ B.\ A \preceq B \longrightarrow \varphi\ A \preceq \varphi\ B\ \langle proof \rangle$
lemma *PD2*: $Der\text{-}1b\ \varphi \Longrightarrow \forall A\ B.\ A \preceq B \longrightarrow \varphi^d\ A \preceq \varphi^d\ B\ \langle proof \rangle$
lemma *PD3*: $Der\text{-}1b\ \varphi \Longrightarrow \forall A\ B.\ \varphi(A \wedge B) \preceq \varphi\ A \wedge \varphi\ B\ \langle proof \rangle$
lemma *PD4*: $Der\text{-}1\ \varphi \Longrightarrow \forall A\ B.\ (\varphi\ A \leftharpoonup \varphi\ B) \preceq \varphi(A \leftharpoonup B)\ \langle proof \rangle$
lemma *PD5*: $Der\text{-}4\ \varphi \Longrightarrow \forall A.\ \varphi(\varphi(-(\varphi\ A))) \preceq \varphi(-(\varphi\ A))\ \langle proof \rangle$

Observe that the lemmas below require Der-2 as premise:

lemma *PD6*: $Der\text{-}1\ \varphi \Longrightarrow Der\text{-}2\ \varphi \Longrightarrow \forall A.\ \varphi^d\ A \preceq \varphi\ A\ \langle proof \rangle$
lemma *PD7*: $Der\text{-}1\ \varphi \Longrightarrow Der\text{-}2\ \varphi \Longrightarrow \forall A.\ \varphi(\varphi^d\ A) \preceq \varphi(\varphi\ A)\ \langle proof \rangle$
lemma *PD8*: $Der\text{-}1\ \varphi \Longrightarrow Der\text{-}2\ \varphi \Longrightarrow Der\text{-}4\ \varphi \Longrightarrow \forall A.\ \varphi(\varphi^d\ A) \preceq \varphi\ A\ \langle proof \rangle$
lemma *PD9*: $Der\text{-}1\ \varphi \Longrightarrow Der\text{-}2\ \varphi \Longrightarrow Der\text{-}4\ \varphi \Longrightarrow \forall A.\ \varphi^d\ A \preceq \varphi^d(\varphi\ A)\ \langle proof \rangle$
lemma *PD10*: $Der\text{-}1\ \varphi \Longrightarrow Der\text{-}2\ \varphi \Longrightarrow Der\text{-}4\ \varphi \Longrightarrow \forall A.\ \varphi^d\ A \preceq \varphi(\varphi^d\ A)\ \langle proof \rangle$
lemma *PD11*: $Der\text{-}1\ \varphi \Longrightarrow Der\text{-}2\ \varphi \Longrightarrow Der\text{-}4\ \varphi \Longrightarrow \forall A.\ -(\varphi\ A) \preceq \varphi(-(\varphi\ A))\ \langle proof \rangle$
lemma *PD12*: $Der\text{-}1\ \varphi \Longrightarrow Der\text{-}2\ \varphi \Longrightarrow Der\text{-}4\ \varphi \Longrightarrow \forall A.\ (\varphi^d\ A) \wedge (\varphi\ A) \approx \varphi^d(A \wedge (\varphi\ A))\ \langle proof \rangle$

The conditions below can serve to axiomatize a derivative operator. Different authors consider different sets of conditions. We define below some corresponding to Zarycki [14], Kuratowski [8] [13], McKinsey & Tarski [11], and Esakia [5], respectively.

abbreviation $\mathfrak{D}z\ \varphi\ \equiv Der\text{-}1\ \varphi \wedge Der\text{-}2\ \varphi \wedge Der\text{-}3\ \varphi \wedge Der\text{-}4\ \varphi \wedge Der\text{-}5\ \varphi$
abbreviation $\mathfrak{D}k\ \varphi\ \equiv Der\text{-}1\ \varphi \wedge Der\text{-}2\ \varphi \wedge Der\text{-}3\ \varphi \wedge Der\text{-}4\ \varphi$
abbreviation $\mathfrak{D}mt\ \varphi \equiv Der\text{-}1\ \varphi \qquad \wedge Der\text{-}3\ \varphi \wedge Der\text{-}4\ \varphi$
abbreviation $\mathfrak{D}e\ \varphi\ \equiv Der\text{-}1\ \varphi \qquad \wedge Der\text{-}3\ \varphi \wedge Der\text{-}4e\ \varphi$

Our 'default' derivative operator will coincide with $\mathfrak{D}k$ from Kuratowski (also Zarycki). However, for proving theorems we will employ the weaker variant Der-4e instead of Der-4 whenever possible. We start by defining a dual operator and verifying some dualities; we then define conversion operators. Observe that conditions Der-2 and Der-5 are not used in the rest of this subsection. Der-2 will be required later when working with the coherence operator.

abbreviation $\mathfrak{D}\ \varphi \equiv \mathfrak{D}k\ \varphi$
abbreviation $\Sigma\ \varphi \equiv Int\text{-}1\ \varphi \wedge Int\text{-}3\ \varphi \wedge Int\text{-}4'\ \varphi \wedge Int\text{-}5\ \varphi$ — 'dual-derivative' operator

lemma *SD-dual1*: $\Sigma(\varphi) \Longrightarrow \mathfrak{D}(\varphi^d)\ \langle proof \rangle$
lemma *SD-dual2*: $\Sigma(\varphi^d) \Longrightarrow \mathfrak{D}(\varphi)\ \langle proof \rangle$
lemma *DS-dual1*: $\mathfrak{D}(\varphi) \Longrightarrow \Sigma(\varphi^d)\ \langle proof \rangle$
lemma *DS-dual2*: $\mathfrak{D}(\varphi^d) \Longrightarrow \Sigma(\varphi)\ \langle proof \rangle$
lemma *DS-dual*: $\mathfrak{D}(\varphi) = \Sigma(\varphi^d)\ \ \langle proof \rangle$

Closure operator as derived from derivative.

definition $Cl\text{-}der::(\sigma{\Rightarrow}\sigma){\Rightarrow}(\sigma{\Rightarrow}\sigma)\ (\mathcal{C}_D)$ **where** $\mathcal{C}_D\ \mathcal{D} \equiv \lambda A.\ A \vee \mathcal{D}(A)$

Verify properties:

lemma *CD1a*: $Der\text{-}1a\ \mathcal{D} \Longrightarrow Cl\text{-}1a\ (\mathcal{C}_D\ \mathcal{D})\ \langle proof \rangle$
lemma *CD1b*: $Der\text{-}1b\ \mathcal{D} \Longrightarrow Cl\text{-}1b\ (\mathcal{C}_D\ \mathcal{D})\ \langle proof \rangle$
lemma *CD1* : $Der\text{-}1\ \mathcal{D} \Longrightarrow Cl\text{-}1\ (\mathcal{C}_D\ \mathcal{D})\ \langle proof \rangle$
lemma *CD2*: $Cl\text{-}2\ (\mathcal{C}_D\ \mathcal{D})\ \langle proof \rangle$
lemma *CD3*: $Der\text{-}3\ \mathcal{D} \Longrightarrow Der\text{-}3\ (\mathcal{C}_D\ \mathcal{D})\ \langle proof \rangle$
lemma *CD4a*: $Der\text{-}1a\ \mathcal{D} \Longrightarrow Der\text{-}4e\ \mathcal{D} \Longrightarrow Cl\text{-}4\ (\mathcal{C}_D\ \mathcal{D})\ \langle proof \rangle$
lemma $Der\text{-}1b\ \mathcal{D} \Longrightarrow Der\text{-}4\ \mathcal{D} \Longrightarrow Cl\text{-}4\ (\mathcal{C}_D\ \mathcal{D})$ **nitpick** $\langle proof \rangle$
lemma *CD4*: $Der\text{-}1\ \mathcal{D} \Longrightarrow Der\text{-}4e\ \mathcal{D} \Longrightarrow Cl\text{-}4\ (\mathcal{C}_D\ \mathcal{D})\ \langle proof \rangle$

Interior operator as derived from (dual) derivative.

definition *Int-der*::$(\sigma{\Rightarrow}\sigma){\Rightarrow}(\sigma{\Rightarrow}\sigma)$ $(\mathcal{I}_D)$ **where** $\mathcal{I}_D\ \mathcal{D} \equiv \lambda A.\ A \wedge \mathcal{D}^d(A)$

Verify definition:

lemma *Int-der-def2*: $\mathcal{I}_D\ \mathcal{D} = (\lambda\varphi.\ \varphi \leftharpoonup \mathcal{D}(-\varphi))$ $\langle proof \rangle$
lemma *dual-der1*: $\mathcal{C}_D\ \mathcal{D} \equiv (\mathcal{I}_D\ \mathcal{D})^d$ $\langle proof \rangle$
lemma *dual-der2*: $\mathcal{I}_D\ \mathcal{D} \equiv (\mathcal{C}_D\ \mathcal{D})^d$ $\langle proof \rangle$

Verify properties:

lemma *ID1*: *Der-1* $\mathcal{D} \Longrightarrow$ *Int-1* $(\mathcal{I}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *ID1a*: *Der-1a* $\mathcal{D} \Longrightarrow$ *Int-1b* $(\mathcal{I}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *ID1b*: *Der-1b* $\mathcal{D} \Longrightarrow$ *Int-1a* $(\mathcal{I}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *ID2*: *Int-2* $(\mathcal{I}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *ID3*: *Der-3* $\mathcal{D} \Longrightarrow$ *Int-3* $(\mathcal{I}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *ID4*: *Der-1* $\mathcal{D} \Longrightarrow$ *Der-4e* $\mathcal{D} \Longrightarrow$ *Int-4* $(\mathcal{I}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *ID4a*: *Der-1a* $\mathcal{D} \Longrightarrow$ *Der-4e* $\mathcal{D} \Longrightarrow$ *Int-4* $(\mathcal{I}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *Der-1b* $\mathcal{D} \Longrightarrow$ *Der-4* $\mathcal{D} \Longrightarrow$ *Int-4* $(\mathcal{I}_D\ \mathcal{D})$ **nitpick** $\langle proof \rangle$

Border operator as derived from (dual) derivative.

definition *Br-der*::$(\sigma{\Rightarrow}\sigma){\Rightarrow}(\sigma{\Rightarrow}\sigma)$ $(\mathcal{B}_D)$ **where** $\mathcal{B}_D\ \mathcal{D} \equiv \lambda A.\ A \leftharpoonup \mathcal{D}^d(A)$

Verify definition:

lemma *Br-der-def2*: $\mathcal{B}_D\ \mathcal{D} = (\lambda A.\ A \wedge \mathcal{D}(-A))$ $\langle proof \rangle$

Verify properties:

lemma *BD1*: *Der-1* $\mathcal{D} \Longrightarrow$ *Br-1* $(\mathcal{B}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *BD2*: *Der-3* $\mathcal{D} \Longrightarrow$ *Br-2* $(\mathcal{B}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *BD3*: *Der-1b* $\mathcal{D} \Longrightarrow$ *Der-4e* $\mathcal{D} \Longrightarrow$ *Br-3* $(\mathcal{B}_D\ \mathcal{D})$ $\langle proof \rangle$

Frontier operator as derived from derivative.

definition *Fr-der*::$(\sigma{\Rightarrow}\sigma){\Rightarrow}(\sigma{\Rightarrow}\sigma)$ $(\mathcal{F}_D)$ **where** $\mathcal{F}_D\ \mathcal{D} \equiv \lambda A.\ (A \leftharpoonup \mathcal{D}^d(A)) \vee (\mathcal{D}(A) \leftharpoonup A)$

Verify definition:

lemma *Fr-der-def2*: $\mathcal{F}_D\ \mathcal{D} = (\lambda A.\ (A \vee \mathcal{D}(A)) \wedge (-A \vee \mathcal{D}(-A)))$ $\langle proof \rangle$

Verify properties:

lemma *FD1a*: *Der-1a* $\mathcal{D} \Longrightarrow$ *Fr-1a*$(\mathcal{F}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *FD1b*: *Der-1b* $\mathcal{D} \Longrightarrow$ *Fr-1b*$(\mathcal{F}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *FD1*: *Der-1* $\mathcal{D} \Longrightarrow$ *Fr-1*$(\mathcal{F}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *FD2*: *Fr-2*$(\mathcal{F}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *FD3*: *Der-3* $\mathcal{D} \Longrightarrow$ *Fr-3*$(\mathcal{F}_D\ \mathcal{D})$ $\langle proof \rangle$
lemma *FD4*: *Der-1* $\mathcal{D} \Longrightarrow$ *Der-4e* $\mathcal{D} \Longrightarrow$ *Fr-4*$(\mathcal{F}_D\ \mathcal{D})$ $\langle proof \rangle$

Note that the derivative operation cannot be obtained from interior, closure, border, nor frontier. In this respect the derivative is a fundamental operation.

4.3.2 Infinitary conditions

The corresponding infinitary condition on derivative operators is inherited from the one for closure.

abbreviation *Der-inf* $\varphi \equiv$ *Cl-inf*(φ)

lemma $CD\text{-}inf$: $Der\text{-}inf\ \varphi \Longrightarrow Cl\text{-}inf(\mathcal{C}_D\ \varphi)$ ⟨*proof*⟩
lemma $ID\text{-}inf$: $Der\text{-}inf\ \varphi \Longrightarrow Int\text{-}inf(\mathcal{I}_D\ \varphi)$ ⟨*proof*⟩

This condition is indeed strong enough as to entail closure of some fixed-point predicates under infimum/supremum.

lemma $fp\text{-}ID\text{-}inf\text{-}closed$: $Der\text{-}inf\ \varphi \Longrightarrow infimum\text{-}closed\ (fp\ (\mathcal{I}_D\ \varphi))$ ⟨*proof*⟩
lemma $fp\text{-}CD\text{-}sup\text{-}closed$: $Der\text{-}inf\ \varphi \Longrightarrow supremum\text{-}closed\ (fp\ (\mathcal{C}_D\ \varphi))$ ⟨*proof*⟩

4.3.3 Coherence conditions

We finish this section by introducing the 'coherence' operator (Cantor's 'Koherenz') as discussed by Zarycki in [13]. As happens with the derivative operator, the coherence operator cannot be derived from interior, closure, border, nor frontier.

definition $Kh\text{-}1\ \varphi \equiv ADDI\text{-}b\ \varphi$
definition $Kh\text{-}2\ \varphi \equiv dEXP\ \varphi$
definition $Kh\text{-}3\ \varphi \equiv \forall A.\ \varphi(\varphi^d\ A) \approx \varphi^d(\varphi\ A)$

lemma $PK1$: $Kh\text{-}1\ \varphi \equiv MONO\ \varphi$ ⟨*proof*⟩
lemma $PK2$: $Kh\text{-}1\ \varphi \equiv \forall A\ B.\ \varphi(A \wedge B) \preceq (\varphi\ A) \wedge (\varphi\ B)$ ⟨*proof*⟩
lemma $PK3$: $Kh\text{-}2\ \varphi \Longrightarrow \varphi\ \bot \approx \bot$ ⟨*proof*⟩
lemma $PK4$: $Kh\text{-}1\ \varphi \Longrightarrow\ Kh\text{-}3\ \varphi \Longrightarrow \varphi\ \top \approx \top$ ⟨*proof*⟩
lemma $PK5$: $Kh\text{-}2\ \varphi \Longrightarrow\ \forall A.\ \varphi(-A) \preceq -(\varphi\ A)$ ⟨*proof*⟩
lemma $PK6$: $Kh\text{-}1\ \varphi \Longrightarrow Kh\text{-}2\ \varphi \Longrightarrow\ \forall A\ B.\ \varphi(A \leftharpoonup B) \preceq (\varphi\ A) \leftharpoonup (\varphi\ B)$ ⟨*proof*⟩
lemma $PK7$: $Kh\text{-}3\ \varphi \Longrightarrow\ \forall A.\ \varphi(\varphi(-(\varphi\ A))) \approx \varphi(-(\varphi(\varphi^d\ A)))$ ⟨*proof*⟩
lemma $PK8$: $Kh\text{-}3\ \varphi \Longrightarrow\ \forall A.\ \varphi(-(\varphi(\varphi\ A))) \approx \varphi^d(\varphi(-(\varphi\ A)))$ ⟨*proof*⟩

Coherence operator as derived from derivative (requires conditions Der-2 and Der-4).

definition $Kh\text{-}der::(\sigma{\Rightarrow}\sigma){\Rightarrow}(\sigma{\Rightarrow}\sigma)\ (\mathcal{K}_D)$ **where** $\mathcal{K}_D\ \mathcal{D} \equiv \lambda A.\ A \wedge (\mathcal{D}\ A)$

Verify properties:

lemma $KD1$: $Der\text{-}1\ \varphi \Longrightarrow Kh\text{-}1\ (\mathcal{K}_D\ \varphi)$ ⟨*proof*⟩
lemma $KD2$: $Kh\text{-}2\ (\mathcal{K}_D\ \varphi)$ ⟨*proof*⟩
lemma $KD3$: $Der\text{-}1\ \varphi \Longrightarrow Der\text{-}2\ \varphi \Longrightarrow Der\text{-}4\ \varphi\ \Longrightarrow Kh\text{-}3\ (\mathcal{K}_D\ \varphi)$ ⟨*proof*⟩

end
theory *topo-alexandrov*
  imports *sse-operation-positive-quantification*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

5 Generalized specialization orderings and Alexandrov topologies

A topology is called 'Alexandrov' (after the Russian mathematician Pavel Alexandrov) if the intersection (resp. union) of any (finite or infinite) family of open (resp. closed) sets is open (resp. closed); in algebraic terms, this means that the set of fixed points of the interior (closure) operation is closed under infinite meets (joins). Another common algebraic formulation requires the closure (interior) operation to satisfy the infinitary variants of additivity (multiplicativity), i.e. iADDI (iMULT) as introduced before.

In the literature, the well-known Kuratowski conditions for the closure (resp. interior) operation are assumed, namely: ADDI, EXP, NOR, IDEM (resp. MULT, dEXP, dNOR, IDEM). This makes both formulations equivalent. However, this is not the case in general if those conditions become negotiable.

Alexandrov topologies have interesting properties relating them to the semantics of modal logic. Assuming Kuratowski conditions, Alexandrov topological operations defined on subsets of S are in one-to-one correspondence with preorders on S; in topological terms, Alexandrov topologies are uniquely determined by their specialization preorders. Since we do not presuppose any Kuratowski conditions to begin with, the preorders in question are in general not even transitive. Here we just call them 'specialization relations'. We will still call (generalized) closure/interior-like operations as such (for lack of a better name). We explore minimal conditions under which some relevant results for the semantics of modal logic obtain.

5.1 Specialization relations

Specialization relations (among worlds/points) are particular cases of propositional functions with type $w \Rightarrow \sigma$.

Define some relevant properties of relations:

abbreviation *serial* $R \equiv \forall x.\ \exists y.\ R\ x\ y$
abbreviation *reflexive* $R \equiv \forall x.\ R\ x\ x$
abbreviation *transitive* $R \equiv \forall x\ y\ z.\ R\ x\ y \wedge R\ y\ z \longrightarrow R\ x\ z$
abbreviation *antisymmetric* $R \equiv \forall x\ y.\ R\ x\ y \wedge R\ y\ x \longrightarrow x = y$
abbreviation *symmetric* $R \equiv \forall x\ y.\ R\ x\ y \longrightarrow R\ y\ x$

Closure/interior operations can be derived from an arbitrary relation as operations returning down-/up-sets.

definition *Cl-rel*::$(w \Rightarrow \sigma) \Rightarrow (\sigma \Rightarrow \sigma)$ $(\mathcal{C}_R)$ **where** $\mathcal{C}_R\ R \equiv \lambda A.\ \lambda w.\ \exists v.\ R\ w\ v \wedge A\ v$
definition *Int-rel*::$(w \Rightarrow \sigma) \Rightarrow (\sigma \Rightarrow \sigma)$ $(\mathcal{I}_R)$ **where** $\mathcal{I}_R\ R \equiv \lambda A.\ \lambda w.\ \forall v.\ R\ w\ v \longrightarrow A\ v$

Duality between interior and closure follows directly:

lemma *dual-rel1*: $\forall A.\ (\mathcal{C}_R\ R)\ A \approx (\mathcal{I}_R\ R)^d\ A$ ⟨*proof*⟩
lemma *dual-rel2*: $\forall A.\ (\mathcal{I}_R\ R)\ A \approx (\mathcal{C}_R\ R)^d\ A$ ⟨*proof*⟩

We explore minimal conditions of the specialization relation under which some operation's conditions obtain.

lemma *rC1*: *ADDI* $(\mathcal{C}_R\ R)$ ⟨*proof*⟩
lemma *rC1i*:*iADDI* $(\mathcal{C}_R\ R)$ ⟨*proof*⟩
lemma *rC2*: *reflexive* $R \longrightarrow$ *EXP* $(\mathcal{C}_R\ R)$ ⟨*proof*⟩
lemma *rC3*: *NOR* $(\mathcal{C}_R\ R)$ ⟨*proof*⟩
lemma *rC4*: *reflexive* $R \wedge$ *transitive* $R \longrightarrow$ *IDEM* $(\mathcal{C}_R\ R)$ ⟨*proof*⟩
lemma *rC-Barcan*: $\forall \pi.\ (\mathcal{C}_R\ R)(\exists x.\ \pi\ x) \preceq (\exists x.\ (\mathcal{C}_R\ R)(\pi\ x))$ ⟨*proof*⟩

lemma *rI1*: *MULT* $(\mathcal{I}_R\ R)$ ⟨*proof*⟩
lemma *rI1i*:*iMULT* $(\mathcal{I}_R\ R)$ ⟨*proof*⟩
lemma *rI2*: *reflexive* $R \Longrightarrow$ *dEXP* $(\mathcal{I}_R\ R)$ ⟨*proof*⟩
lemma *rI3*: *dNOR* $(\mathcal{I}_R\ R)$ ⟨*proof*⟩
lemma *rI4*: *reflexive* $R \Longrightarrow$ *transitive* $R \Longrightarrow$ *IDEM* $(\mathcal{I}_R\ R)$ ⟨*proof*⟩
lemma *rI-Barcan*: $\forall \pi.\ (\forall x.\ (\mathcal{I}_R\ R)(\pi\ x)) \preceq (\mathcal{I}_R\ R)(\forall x.\ \pi\ x)$ ⟨*proof*⟩

A specialization relation can be derived from a given operation (intended as a closure-like operation).

definition *sp-rel*::$(\sigma \Rightarrow \sigma) \Rightarrow (w \Rightarrow \sigma)$ $(\mathcal{R}^C)$ **where** $\mathcal{R}^C\ \mathcal{C} \equiv \lambda w\ v.\ \mathcal{C}\ (\lambda u.\ u{=}v)\ w$

Preorder properties of the specialization relation follow directly from the corresponding operation's conditions.

lemma *sp-rel-reflex*: $EXP\ \mathcal{C} \Longrightarrow reflexive\ (\mathcal{R}^C\ \mathcal{C})\ \langle proof \rangle$
lemma *sp-rel-trans*: $MONO\ \mathcal{C} \Longrightarrow IDEM\ \mathcal{C} \Longrightarrow transitive\ (\mathcal{R}^C\ \mathcal{C})\ \langle proof \rangle$

However, we can obtain finite countermodels for antisymmetry and symmetry given all relevant conditions. We will revisit this issue later and examine their relation with the topological separation axioms T0 and T1 resp.

lemma $iADDI\ \mathcal{C} \Longrightarrow EXP\ \mathcal{C} \Longrightarrow NOR\ \mathcal{C} \Longrightarrow IDEM\ \mathcal{C} \Longrightarrow antisymmetric\ (\mathcal{R}^C\ \mathcal{C})$ **nitpick** $\langle proof \rangle$
lemma $iADDI\ \mathcal{C} \Longrightarrow EXP\ \mathcal{C} \Longrightarrow NOR\ \mathcal{C} \Longrightarrow IDEM\ \mathcal{C} \Longrightarrow symmetric\ (\mathcal{R}^C\ \mathcal{C})$ **nitpick** $\langle proof \rangle$

5.2 Alexandrov topology

As mentioned previously, Alexandrov closure (and by duality interior) operations correspond to specialization relations. It is worth mentioning that in Alexandrov topologies every point has a minimal/smallest neighborhood, namely the set of points related to it by the specialization (aka. accessibility) relation. Alexandrov spaces are thus also called 'finitely generated'. We examine below minimal conditions under which these relations obtain.

lemma *sp-rel-a*: $MONO\ \mathcal{C} \Longrightarrow \forall A.\ (\mathcal{C}_R\ (\mathcal{R}^C\ \mathcal{C}))\ A \preceq \mathcal{C}\ A\ \langle proof \rangle$
lemma *sp-rel-b*: $iADDI\text{-}a\ \mathcal{C} \Longrightarrow \forall A.\ (\mathcal{C}_R\ (\mathcal{R}^C\ \mathcal{C}))\ A \succeq \mathcal{C}\ A\ \langle proof \rangle$
lemma *sp-rel*: $iADDI\ \mathcal{C} \Longrightarrow \forall A.\ \mathcal{C}\ A \approx (\mathcal{C}_R\ (\mathcal{R}^C\ \mathcal{C}))\ A\ \langle proof \rangle$

It is instructive to expand the definitions in the above lemma:

lemma $iADDI\ \mathcal{C} \Longrightarrow\ \forall A.\ \forall w.\ (\mathcal{C}\ A)\ w \longleftrightarrow (\exists v.\ A\ v \land (\mathcal{C}\ (\lambda u.\ u{=}v))\ w)\ \langle proof \rangle$

We now turn to the more traditional characterization of Alexandrov topologies in terms of closure under infinite joins/meets.

Fixed points of operations satisfying ADDI (MULT) are not in general closed under infinite joins (meets). For the given conditions countermodels are expected to be infinite. We (sanity) check that nitpick cannot find any.

lemma $ADDI(\varphi) \Longrightarrow supremum\text{-}closed\ (fp\ \varphi)\ \langle proof \rangle$
lemma $MULT(\varphi) \Longrightarrow infimum\text{-}closed\ (fp\ \varphi)\ \langle proof \rangle$

By contrast, we can show that this obtains if assuming the corresponding infinitary variants (iADDI/iMULT).

lemma $iADDI(\varphi) \Longrightarrow supremum\text{-}closed\ (fp\ \varphi)\ \langle proof \rangle$
lemma $iMULT(\varphi) \Longrightarrow infimum\text{-}closed\ (fp\ \varphi)\ \langle proof \rangle$

As shown above, closure (interior) operations derived from relations readily satisfy iADDI (iMULT), being thus closed under infinite joins (meets).

lemma $supremum\text{-}closed\ (fp\ (\mathcal{C}_R\ R))\ \langle proof \rangle$
lemma $infimum\text{-}closed\ (fp\ (\mathcal{I}_R\ R))\ \langle proof \rangle$

5.3 (Anti)symmetry and the separation axioms T0 and T1

We can now revisit the relationship between (anti)symmetry and the separation axioms T1 and T0.

T0: any two distinct points in the space can be separated by an open set (i.e. containing one point and not the other).

abbreviation $T0\text{-}sep\ \mathcal{C} \equiv \forall\, w\ v.\ w \neq v \longrightarrow (\exists\, G.\ (fp\ \mathcal{C}^d)(G) \wedge (G\ w \neq G\ v))$

T1: any two distinct points can be separated by (two not necessarily disjoint) open sets, i.e. all singletons are closed.

abbreviation $T1\text{-}sep\ \mathcal{C} \equiv \forall\, w.\ (fp\ \mathcal{C})(\lambda u.\ u = w)$

We can (sanity) check that T1 entails T0 but not viceversa.

lemma $T0\text{-}sep\ \mathcal{C} \Longrightarrow T1\text{-}sep\ \mathcal{C}$ **nitpick** ⟨*proof*⟩
lemma $T1\text{-}sep\ \mathcal{C} \Longrightarrow T0\text{-}sep\ \mathcal{C}$ ⟨*proof*⟩

Under appropriate conditions, T0-separation corresponds to antisymmetry of the specialization relation (here an ordering).

lemma $T0\text{-}sep\ \mathcal{C} \longleftrightarrow antisymmetric\ (\mathcal{R}^C\ \mathcal{C})$ **nitpick** ⟨*proof*⟩
lemma *T0-antisymm-a*: $MONO\ \mathcal{C} \Longrightarrow T0\text{-}sep\ \mathcal{C} \longrightarrow antisymmetric\ (\mathcal{R}^C\ \mathcal{C})$ ⟨*proof*⟩
lemma *T0-antisymm-b*: $EXP\ \mathcal{C} \Longrightarrow IDEM\ \mathcal{C} \Longrightarrow antisymmetric\ (\mathcal{R}^C\ \mathcal{C}) \longrightarrow T0\text{-}sep\ \mathcal{C}$ ⟨*proof*⟩
lemma *T0-antisymm*: $MONO\ \mathcal{C} \Longrightarrow EXP\ \mathcal{C} \Longrightarrow IDEM\ \mathcal{C} \Longrightarrow T0\text{-}sep\ \mathcal{C} = antisymmetric\ (\mathcal{R}^C\ \mathcal{C})$ ⟨*proof*⟩

Also, under the appropriate conditions, T1-separation corresponds to symmetry of the specialization relation.

lemma *T1-symm-a*: $T1\text{-}sep\ \mathcal{C} \longrightarrow symmetric\ (\mathcal{R}^C\ \mathcal{C})$ ⟨*proof*⟩
lemma *T1-symm-b*: $MONO\ \mathcal{C} \Longrightarrow EXP\ \mathcal{C} \Longrightarrow T0\text{-}sep\ \mathcal{C} \Longrightarrow symmetric\ (\mathcal{R}^C\ \mathcal{C}) \longrightarrow T1\text{-}sep\ \mathcal{C}$ ⟨*proof*⟩
lemma *T1-symm*: $MONO\ \mathcal{C} \Longrightarrow EXP\ \mathcal{C} \Longrightarrow T0\text{-}sep\ \mathcal{C} \Longrightarrow symmetric\ (\mathcal{R}^C\ \mathcal{C}) = T1\text{-}sep\ \mathcal{C}$ ⟨*proof*⟩

end
theory *topo-frontier-algebra*
  imports *topo-operators-basic*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

6 Frontier Algebra

The closure of a set A ($\mathcal{C}(A)$) can be seen as the set A augmented by (i) its boundary points, or (ii) its accumulation/limit points. In this section we explore the first variant by drawing on the notion of a frontier algebra, defined in an analogous fashion as the well-known closure and interior algebras.

Declares a primitive (unconstrained) frontier (aka. boundary) operation and defines others from it.

consts $\mathcal{F}{::}\sigma{\Rightarrow}\sigma$
abbreviation $\mathcal{I} \equiv \mathcal{I}_F\ \mathcal{F}$ — interior
abbreviation $\mathcal{C} \equiv \mathcal{C}_F\ \mathcal{F}$ — closure
abbreviation $\mathcal{B} \equiv \mathcal{B}_F\ \mathcal{F}$ — border

6.1 Basic properties

Verifies minimal conditions under which operators resulting from conversion functions coincide.

lemma *ICdual*: $Fr\text{-}2\ \mathcal{F} \Longrightarrow \mathcal{I} \equiv \mathcal{C}^d$ ⟨*proof*⟩
lemma *ICdual'*: $Fr\text{-}2\ \mathcal{F} \Longrightarrow \mathcal{C} \equiv \mathcal{I}^d$ ⟨*proof*⟩
lemma *BI-rel*: $\mathcal{B} \equiv \mathcal{B}_I\ \mathcal{I}$ ⟨*proof*⟩
lemma *IB-rel*: $\mathcal{I} \equiv \mathcal{I}_B\ \mathcal{B}$ ⟨*proof*⟩
lemma *BC-rel*: $Fr\text{-}2\ \mathcal{F} \Longrightarrow \mathcal{B} \equiv \mathcal{B}_C\ \mathcal{C}$ ⟨*proof*⟩

lemma *CB-rel*: *Fr-2* $\mathcal{F} \Longrightarrow \mathcal{C} \equiv \mathcal{C}_B\ \mathcal{B}$ ⟨*proof*⟩
lemma *FI-rel*: *Fr-2* $\mathcal{F} \Longrightarrow \mathcal{F} \equiv \mathcal{F}_I\ \mathcal{I}$ ⟨*proof*⟩
lemma *FC-rel*: *Fr-2* $\mathcal{F} \Longrightarrow \mathcal{F} \equiv \mathcal{F}_C\ \mathcal{C}$ ⟨*proof*⟩
lemma *FB-rel*: *Fr-2* $\mathcal{F} \Longrightarrow \mathcal{F} \equiv \mathcal{F}_B\ \mathcal{B}$ ⟨*proof*⟩

Fixed-point and other operators are interestingly related.

lemma *fp1*: $\mathcal{I}^{fp} \equiv \mathcal{B}^c$ ⟨*proof*⟩
lemma *fp2*: $\mathcal{B}^{fp} \equiv \mathcal{I}^c$ ⟨*proof*⟩
lemma *fp3*: *Fr-2* $\mathcal{F} \Longrightarrow \mathcal{C}^{fp} \equiv \mathcal{B}^d$ ⟨*proof*⟩
lemma *fp4*: *Fr-2* $\mathcal{F} \Longrightarrow (\mathcal{B}^d)^{fp} \equiv \mathcal{C}$ ⟨*proof*⟩
lemma *fp5*: $\mathcal{F}^{fp} \equiv \mathcal{B} \sqcup (\mathcal{C}^c)$ ⟨*proof*⟩

Different inter-relations (some redundant ones are kept to help the provers).

lemma *monI*: *Fr-1b* $\mathcal{F} \Longrightarrow MONO(\mathcal{I})$ ⟨*proof*⟩
lemma *monC*: *Fr-6* $\mathcal{F} \Longrightarrow MONO(\mathcal{C})$ ⟨*proof*⟩
lemma *pB1*: $\forall A.\ \mathcal{B}\ A \approx A \leftharpoonup \mathcal{I}\ A$ ⟨*proof*⟩
lemma *pB2*: $\forall A.\ \mathcal{B}\ A \approx A \wedge \mathcal{F}\ A$ ⟨*proof*⟩
lemma *pB3*: *Fr-2* $\mathcal{F} \Longrightarrow \forall A.\ \mathcal{B}(-A) \approx -A \wedge \mathcal{F}\ A$ ⟨*proof*⟩
lemma *pB4*: *Fr-2* $\mathcal{F} \Longrightarrow \forall A.\ \mathcal{B}(-A) \approx -A \wedge \mathcal{C}\ A$ ⟨*proof*⟩
lemma *pB5*: *Fr-1b* $\mathcal{F} \Longrightarrow$ *Fr-2* $\mathcal{F} \Longrightarrow \forall A.\ \mathcal{B}(\mathcal{C}\ A) \preceq (\mathcal{B}\ A) \vee \mathcal{B}(-A)$ ⟨*proof*⟩
lemma *pF1*: $\forall A.\ \mathcal{F}\ A \approx \mathcal{C}\ A \leftharpoonup \mathcal{I}\ A$ ⟨*proof*⟩
lemma *pF2*: *Fr-2* $\mathcal{F} \Longrightarrow \forall A.\ \mathcal{F}\ A \approx \mathcal{C}\ A \wedge \mathcal{C}(-A)$ ⟨*proof*⟩
lemma *pF3*: *Fr-2* $\mathcal{F} \Longrightarrow \forall A.\ \mathcal{F}\ A \approx \mathcal{B}\ A \vee \mathcal{B}(-A)$ ⟨*proof*⟩
lemma *pF4*: *Fr-1* $\mathcal{F} \Longrightarrow$ *Fr-2* $\mathcal{F} \Longrightarrow$ *Fr-4*$(\mathcal{F}) \Longrightarrow \forall A.\ \mathcal{F}(\mathcal{I}\ A) \preceq \mathcal{F}\ A$ ⟨*proof*⟩
lemma *pF5*: *Fr-1* $\mathcal{F} \Longrightarrow$ *Fr-2* $\mathcal{F} \Longrightarrow$ *Fr-4* $\mathcal{F} \Longrightarrow \forall A.\ \mathcal{F}(\mathcal{C}\ A) \preceq \mathcal{F}\ A$ ⟨*proof*⟩
lemma *pA1*: $\forall A.\ A \approx \mathcal{I}\ A \vee \mathcal{B}\ A$ ⟨*proof*⟩
lemma *pA2*: *Fr-2* $\mathcal{F} \Longrightarrow \forall A.\ A \approx \mathcal{C}\ A \leftharpoonup \mathcal{B}(-A)$ ⟨*proof*⟩
lemma *pC1*: *Fr-2* $\mathcal{F} \Longrightarrow \forall A.\ \mathcal{C}\ A \approx A \vee \mathcal{B}(-A)$ ⟨*proof*⟩
lemma *pC2*: $\forall A.\ \mathcal{C}\ A \approx A \vee \mathcal{F}\ A$ ⟨*proof*⟩
lemma *pI1*: $\forall A.\ \mathcal{I}\ A \approx A \leftharpoonup \mathcal{B}\ A$ ⟨*proof*⟩
lemma *pI2*: $\forall A.\ \mathcal{I}\ A \approx A \leftharpoonup \mathcal{F}\ A$ ⟨*proof*⟩

lemma *IC-imp*: *Fr-1* $\mathcal{F} \Longrightarrow$ *Fr-2* $\mathcal{F} \Longrightarrow$ *Fr-3* $\mathcal{F} \Longrightarrow \forall A\ B.\ \mathcal{I}(A \rightarrow B) \preceq \mathcal{C}\ A \rightarrow \mathcal{C}\ B$ ⟨*proof*⟩

Defines some fixed-point predicates and prove some properties.

abbreviation *openset* (*Op*) **where** *Op* $A \equiv fp\ \mathcal{I}\ A$
abbreviation *closedset* (*Cl*) **where** *Cl* $A \equiv fp\ \mathcal{C}\ A$
abbreviation *borderset* (*Br*) **where** *Br* $A \equiv fp\ \mathcal{B}\ A$
abbreviation *frontierset* (*Fr*) **where** *Fr* $A \equiv fp\ \mathcal{F}\ A$

lemma *Int-Open*: *Fr-1a* $\mathcal{F} \Longrightarrow$ *Fr-2* $\mathcal{F} \Longrightarrow$ *Fr-4* $\mathcal{F} \Longrightarrow \forall A.\ Op(\mathcal{I}\ A)$ ⟨*proof*⟩
lemma *Cl-Closed*: *Fr-1a* $\mathcal{F} \Longrightarrow$ *Fr-2* $\mathcal{F} \Longrightarrow$ *Fr-4* $\mathcal{F} \Longrightarrow \forall A.\ Cl(\mathcal{C}\ A)$ ⟨*proof*⟩
lemma *Br-Border*: *Fr-1b* $\mathcal{F} \Longrightarrow \forall A.\ Br(\mathcal{B}\ A)$ ⟨*proof*⟩

In contrast, there is no analogous fixed-point result for frontier:

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall A.\ Fr(\mathcal{F}\ A)$ **nitpick** ⟨*proof*⟩

lemma *OpCldual*: *Fr-2* $\mathcal{F} \Longrightarrow \forall A.\ Cl\ A \longleftrightarrow Op(-A)$ ⟨*proof*⟩
lemma *ClOpdual*: *Fr-2* $\mathcal{F} \Longrightarrow \forall A.\ Op\ A \longleftrightarrow Cl(-A)$ ⟨*proof*⟩
lemma *Fr-ClBr*: $\forall A.\ Fr(A) = (Cl(A) \wedge Br(A))$ ⟨*proof*⟩
lemma *Cl-F*: *Fr-4* $\mathcal{F} \Longrightarrow \forall A.\ Cl(\mathcal{F}\ A)$ ⟨*proof*⟩

6.2 Further properties

The definitions and theorems below are well known in the literature (e.g. [9]). Here we uncover the minimal conditions under which they hold (taking frontier operation as primitive).

lemma *Cl-Bzero*: $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A.\ \mathit{Cl}\ A \longleftrightarrow \mathcal{B}(-A) \approx \bot$ ⟨*proof*⟩
lemma *Op-Bzero*: $\forall A.\ \mathit{Op}\ A \longleftrightarrow (\mathcal{B}\ A) \approx \bot$ ⟨*proof*⟩
lemma *Br-boundary*: $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A.\ \mathit{Br}(A) \longleftrightarrow \mathcal{I}\ A \approx \bot$ ⟨*proof*⟩
lemma *Fr-nowhereDense*: $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A.\ \mathit{Fr}(A) \longrightarrow \mathcal{I}(\mathcal{C}\ A) \approx \bot$ ⟨*proof*⟩
lemma *Cl-FB*: $\forall A.\ \mathit{Cl}\ A \longleftrightarrow \mathcal{F}\ A \approx \mathcal{B}\ A$ ⟨*proof*⟩
lemma *Op-FB*: $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A.\ \mathit{Op}\ A \longleftrightarrow \mathcal{F}\ A \approx \mathcal{B}(-A)$ ⟨*proof*⟩
lemma *Clopen-Fzero*: $\forall A.\ \mathit{Cl}\ A \wedge \mathit{Op}\ A \longleftrightarrow \mathcal{F}\ A \approx \bot$ ⟨*proof*⟩

lemma *Int-sup-closed*: $\mathit{Fr\text{-}1b}\ \mathcal{F} \Longrightarrow \mathit{supremum\text{-}closed}\ (\lambda A.\ \mathit{Op}\ A)$ ⟨*proof*⟩
lemma *Int-meet-closed*: $\mathit{Fr\text{-}1a}\ \mathcal{F} \Longrightarrow \mathit{meet\text{-}closed}\ (\lambda A.\ \mathit{Op}\ A)$ ⟨*proof*⟩
lemma *Int-inf-closed*: $\mathit{Fr\text{-}inf}\ \mathcal{F} \Longrightarrow \mathit{infimum\text{-}closed}\ (\lambda A.\ \mathit{Op}\ A)$ ⟨*proof*⟩
lemma *Cl-inf-closed*: $\mathit{Fr\text{-}6}\ \mathcal{F} \Longrightarrow \mathit{infimum\text{-}closed}\ (\lambda A.\ \mathit{Cl}\ A)$ ⟨*proof*⟩
lemma *Cl-join-closed*: $\mathit{Fr\text{-}1a}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{join\text{-}closed}\ (\lambda A.\ \mathit{Cl}\ A)$ ⟨*proof*⟩
lemma *Cl-sup-closed*: $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}inf}\ \mathcal{F} \Longrightarrow \mathit{supremum\text{-}closed}\ (\lambda A.\ \mathit{Cl}\ A)$ ⟨*proof*⟩
lemma *Br-inf-closed*: $\mathit{Fr\text{-}1b}\ \mathcal{F} \Longrightarrow \mathit{infimum\text{-}closed}\ (\lambda A.\ \mathit{Br}\ A)$ ⟨*proof*⟩
lemma *Fr-inf-closed*: $\mathit{Fr\text{-}1b}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{infimum\text{-}closed}\ (\lambda A.\ \mathit{Fr}\ A)$ ⟨*proof*⟩
lemma *Br-Fr-join*: $\mathit{Fr\text{-}1}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}4}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Br}\ A \wedge \mathit{Fr}\ B \longrightarrow \mathit{Br}(A \vee B)$ ⟨*proof*⟩
lemma *Fr-join-closed*: $\mathit{Fr\text{-}1}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}4}\ \mathcal{F} \Longrightarrow \mathit{join\text{-}closed}\ (\lambda A.\ \mathit{Fr}\ A)$ ⟨*proof*⟩

Introduces a predicate for indicating that two sets are disjoint and proves some properties.

abbreviation $\mathit{Disj}\ A\ B \equiv A \wedge B \approx \bot$

lemma *Disj-comm*: $\forall A\ B.\ \mathit{Disj}\ A\ B \longrightarrow \mathit{Disj}\ B\ A$ ⟨*proof*⟩
lemma *Disj-IF*: $\forall A.\ \mathit{Disj}\ (\mathcal{I}\ A)\ (\mathcal{F}\ A)$ ⟨*proof*⟩
lemma *Disj-B*: $\forall A.\ \mathit{Disj}\ (\mathcal{B}\ A)\ (\mathcal{B}(-A))$ ⟨*proof*⟩
lemma *Disj-I*: $\forall A.\ \mathit{Disj}\ (\mathcal{I}\ A)\ (-A)$ ⟨*proof*⟩
lemma *Disj-BCI*: $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A.\ \mathit{Disj}\ (\mathcal{B}(\mathcal{C}\ A))\ (\mathcal{I}(-A))$ ⟨*proof*⟩
lemma *Disj-CBI*: $\mathit{Fr\text{-}6}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}4}\ \mathcal{F} \Longrightarrow \forall A.\ \mathit{Disj}\ (\mathcal{C}(\mathcal{B}(-A)))\ (\mathcal{I}(-A))$ ⟨*proof*⟩

Introduces a predicate for indicating that two sets are separated and proves some properties.

definition $\mathit{Sep}\ A\ B \equiv \mathit{Disj}\ (\mathcal{C}\ A)\ B \wedge \mathit{Disj}\ (\mathcal{C}\ B)\ A$

lemma *Sep-comm*: $\forall A\ B.\ \mathit{Sep}\ A\ B \longrightarrow \mathit{Sep}\ B\ A$ ⟨*proof*⟩
lemma *Sep-disj*: $\forall A\ B.\ \mathit{Sep}\ A\ B \longrightarrow \mathit{Disj}\ A\ B$ ⟨*proof*⟩
lemma *Sep-I*: $\mathit{Fr\text{-}1}(\mathcal{F}) \Longrightarrow \mathit{Fr\text{-}2}(\mathcal{F}) \Longrightarrow \mathit{Fr\text{-}4}(\mathcal{F}) \Longrightarrow \forall A.\ \mathit{Sep}\ (\mathcal{I}\ A)\ (\mathcal{I}\ (-A))$ ⟨*proof*⟩
lemma *Sep-sub*: $\mathit{Fr\text{-}6}\ \mathcal{F} \Longrightarrow \forall A\ B\ C\ D.\ \mathit{Sep}\ A\ B \wedge C \preceq A \wedge D \preceq B \longrightarrow \mathit{Sep}\ C\ D$ ⟨*proof*⟩
lemma *Sep-Cl-diff*: $\mathit{Fr\text{-}6}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Cl}(A) \wedge \mathit{Cl}(B) \longrightarrow \mathit{Sep}\ (A \leftharpoondown B)\ (B \leftharpoondown A)$ ⟨*proof*⟩
lemma *Sep-Op-diff*: $\mathit{Fr\text{-}1b}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Op}(A) \wedge \mathit{Op}(B) \longrightarrow \mathit{Sep}\ (A \leftharpoondown B)\ (B \leftharpoondown A)$ ⟨*proof*⟩
lemma *Sep-Cl*: $\forall A\ B.\ \mathit{Cl}(A) \wedge \mathit{Cl}(B) \wedge \mathit{Disj}\ A\ B \longrightarrow \mathit{Sep}\ A\ B$ ⟨*proof*⟩
lemma *Sep-Op*: $\mathit{Fr\text{-}1b}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Op}(A) \wedge \mathit{Op}(B) \wedge \mathit{Disj}\ A\ B \longrightarrow \mathit{Sep}\ A\ B$ ⟨*proof*⟩
lemma $\mathit{Fr\text{-}1a}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B\ C.\ \mathit{Sep}\ A\ B \wedge \mathit{Sep}\ A\ C \longrightarrow \mathit{Sep}\ A\ (B \vee C)$ ⟨*proof*⟩

Verifies a neighborhood-based definition of closure.

definition $\mathit{nbhd}\ A\ p \equiv \exists E.\ E \preceq A \wedge \mathit{Op}(E) \wedge (E\ p)$
lemma *nbhd-def2*: $\mathit{Fr\text{-}1}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}4}\ \mathcal{F} \Longrightarrow \forall A\ p.\ (\mathit{nbhd}\ A\ p) = (\mathcal{I}\ A\ p)$ ⟨*proof*⟩

lemma *C-def-lr*: $\mathit{Fr\text{-}1b}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}4}\ \mathcal{F} \Longrightarrow \forall A\ p.\ (\mathcal{C}\ A\ p) \longrightarrow (\forall E.\ (\mathit{nbhd}\ E\ p) \longrightarrow \neg(\mathit{Disj}\ E\ A))$ ⟨*proof*⟩
lemma *C-def-rl*: $\mathit{Fr\text{-}1}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}4}\ \mathcal{F} \Longrightarrow \forall A\ p.\ (\mathcal{C}\ A\ p) \longleftarrow (\forall E.\ (\mathit{nbhd}\ E\ p) \longrightarrow \neg(\mathit{Disj}\ E\ A))$ ⟨*proof*⟩

lemma *C-def*: *Fr-1* $\mathcal{F} \Longrightarrow$ *Fr-2* $\mathcal{F} \Longrightarrow$ *Fr-4* $\mathcal{F} \Longrightarrow \forall A\ p.\ (\mathcal{C}\ A\ p) \longleftrightarrow (\forall E.\ (nbhd\ E\ p) \longrightarrow \neg(Disj\ E\ A))$ ⟨*proof*⟩

Explore the Barcan and converse Barcan formulas.

lemma *Barcan-I*: *Fr-inf* $\mathcal{F} \Longrightarrow \forall P.\ (\forall x.\ \mathcal{I}(P\ x)) \preceq \mathcal{I}(\forall x.\ P\ x)$ ⟨*proof*⟩
lemma *Barcan-C*: *Fr-2* $\mathcal{F} \Longrightarrow$ *Fr-inf* $\mathcal{F} \Longrightarrow \forall P.\ \mathcal{C}(\exists x.\ P\ x) \preceq (\exists x.\ \mathcal{C}(P\ x))$ ⟨*proof*⟩
lemma *CBarcan-I*: *Fr-1b* $\mathcal{F} \Longrightarrow \forall P.\ \mathcal{I}(\forall x.\ P\ x) \preceq (\forall x.\ \mathcal{I}(P\ x))$ ⟨*proof*⟩
lemma *CBarcan-C*: *Fr-6* $\mathcal{F} \Longrightarrow \forall P.\ (\exists x.\ \mathcal{C}(P\ x)) \preceq \mathcal{C}(\exists x.\ P\ x)$ ⟨*proof*⟩

end
theory *topo-negation-conditions*
imports *topo-frontier-algebra sse-operation-negative-quantification*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

7 Frontier-based negation - Semantic conditions

We define a paracomplete and a paraconsistent negation employing the interior and closure operation resp. We take the frontier operator as primitive and explore which semantic conditions are minimally required to obtain some relevant properties of negation.

definition *neg-I*::$\sigma \Rightarrow \sigma$ $(\neg^I)$ **where** $\neg^I\ A \equiv \mathcal{I}(-A)$
definition *neg-C*::$\sigma \Rightarrow \sigma$ $(\neg^C)$ **where** $\neg^C\ A \equiv \mathcal{C}(-A)$
declare *neg-I-def*[*conn*] *neg-C-def*[*conn*]

(We rename the meta-logical HOL negation to avoid terminological confusion)

abbreviation *cneg*::*bool*⇒*bool* (∼- [*40*]*40*) **where** $\sim\varphi \equiv \neg\varphi$

7.1 'Explosion' (ECQ), non-contradiction (LNC) and excluded middle (TND)

TND

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow$ *TNDm* $\neg^I$ **nitpick** ⟨*proof*⟩
lemma *TND-C*: *TND* $\neg^C$ ⟨*proof*⟩

ECQ

lemma *ECQ-I*: *ECQ* $\neg^I$ ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow$ *ECQm* $\neg^C$ **nitpick** ⟨*proof*⟩

LNC

lemma *LNC* $\neg^I$ **nitpick** ⟨*proof*⟩
lemma *LNC-I*: *Fr-2* $\mathcal{F} \Longrightarrow$ *Fr-3* $\mathcal{F} \Longrightarrow$ *LNC* $\neg^I$ ⟨*proof*⟩
lemma *LNC* $\neg^C$ **nitpick** ⟨*proof*⟩
lemma *LNC-C*: *Fr-6* $\mathcal{F} \Longrightarrow$ *LNC* $\neg^C$ ⟨*proof*⟩

Relations between LNC and different ECQ variants (only relevant for paraconsistent negation).

lemma *ECQ* $\neg^C \longrightarrow$ *LNC* $\neg^C$ ⟨*proof*⟩
lemma *ECQw-LNC*: *ECQw* $\neg^C \longrightarrow$ *LNC* $\neg^C$ ⟨*proof*⟩
lemma *ECQm-LNC*: *Fr-1* $\mathcal{F} \Longrightarrow$ *Fr-2* $\mathcal{F} \Longrightarrow$ *ECQm* $\neg^C \longrightarrow$ *LNC* $\neg^C$ ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow$ *LNC* $\neg^C \longrightarrow$ *ECQm* $\neg^C$ **nitpick** ⟨*proof*⟩

Below we show that enforcing all conditions on the frontier operator still leaves room for both boldly paraconsistent and paracomplete models. We use Nitpick to generate a non-trivial model (here a set algebra with 8 elements).

lemma $\mathfrak{F}\ \mathcal{F} \wedge \sim ECQm\ \neg^C$ **nitpick**[*satisfy*,*card w=3*] ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \wedge \sim TNDm\ \neg^I$ **nitpick**[*satisfy*,*card w=3*] ⟨*proof*⟩

7.2 Modus tollens (MT)

MT-I

lemma *MT0-I*: $Fr\text{-}1b\ \mathcal{F} \Longrightarrow MT0\ \neg^I$ ⟨*proof*⟩
lemma *MT1-I*: $Fr\text{-}1b\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow MT1\ \neg^I$ ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow MT2\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma $\sim TND\ \neg^I \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge MT2\ \neg^I$ **nitpick**[*satisfy*,*card w=3*] ⟨*proof*⟩
lemma $\sim TNDm\ \neg^I \wedge Fr\text{-}1a\ \mathcal{F} \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}3\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge MT2\ \neg^I$ **nitpick**[*satisfy*] ⟨*proof*⟩
lemma $\sim TNDm\ \neg^I \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge MT2\ \neg^I$ **nitpick**[*satisfy*,*card w=3*] ⟨*proof*⟩
lemma $\sim TNDm\ \neg^I \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}3\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge MT2\ \neg^I$ **nitpick**[*satisfy*,*card w=3*] ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow MT3\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma $\sim TNDm\ \neg^I \wedge Fr\text{-}1a\ \mathcal{F} \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}3\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge MT3\ \neg^I$ **nitpick**[*satisfy*,*card w=3*] ⟨*proof*⟩
lemma $\sim TNDm\ \neg^I \wedge MT0\ \neg^I \wedge MT1\ \neg^I \wedge MT2\ \neg^I \wedge MT3\ \neg^I$ **nitpick**[*satisfy*,*card w=3*] ⟨*proof*⟩

MT-C

lemma $Fr\text{-}2\ \mathcal{F} \Longrightarrow MT0\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma *MT0-C*: $Fr\text{-}6\ \mathcal{F} \Longrightarrow MT0\ \neg^C$ ⟨*proof*⟩
lemma *MT1-C*: $Fr\text{-}6\ \mathcal{F} \Longrightarrow MT1\ \neg^C$ ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow MT2\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma $\sim ECQm\ \neg^C \wedge \mathfrak{F}\ \mathcal{F} \wedge MT2\ \neg^C$ **nitpick**[*satisfy*,*card w=3*] ⟨*proof*⟩
lemma *MT3-C*: $Fr\text{-}1b\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow MT3\ \neg^C$ ⟨*proof*⟩
lemma $\sim ECQm\ \neg^C \wedge MT0\ \neg^C \wedge MT1\ \neg^C \wedge MT2\ \neg^C \wedge MT3\ \neg^C$ **nitpick**[*satisfy*,*card w=3*] ⟨*proof*⟩

7.3 Contraposition rules (CoP)

CoPw-I

lemma $CoPw\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma *CoPw-I*: $Fr\text{-}1b\ \mathcal{F} \Longrightarrow CoPw\ \neg^I$ ⟨*proof*⟩

CoPw-C

lemma $CoPw\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma *CoPw-C*: $Fr\text{-}6\ \mathcal{F} \Longrightarrow CoPw\ \neg^C$ ⟨*proof*⟩

We can indeed prove that XCoP is entailed by CoP1 (CoP2) in the particular case of a closure-(interior-)based negation.

lemma *CoP1-XCoP*: $CoP1\ \neg^C \longrightarrow XCoP\ \neg^C$ ⟨*proof*⟩
lemma *CoP2-XCoP*: $CoP2\ \neg^I \longrightarrow XCoP\ \neg^I$ ⟨*proof*⟩

CoP1-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow CoP1\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma $\sim TNDm\ \neg^I \wedge \mathfrak{F}\ \mathcal{F} \wedge CoP1\ \neg^I$ **nitpick**[*satisfy*,*card w=3*] ⟨*proof*⟩

CoP1-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow CoP1\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma $\sim ECQ\ \neg^C \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge CoP1\ \neg^C$ **nitpick**[*satisfy*,*card w=3*] ⟨*proof*⟩
lemma $CoP1\ \neg^C \longrightarrow ECQm\ \neg^C$ ⟨*proof*⟩

CoP2-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow CoP2\ \neg^I$ **nitpick** ⟨*proof*⟩

lemma $\sim TND\ \neg^I \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge CoP2\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩
lemma $\sim TND\ \neg^I \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}3\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge CoP2\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩
lemma $CoP2\ \neg^I \longrightarrow TNDm\ \neg^I$ ⟨*proof*⟩

CoP2-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow CoP2\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma $\sim ECQm\ \neg^C \wedge \mathfrak{F}\ \mathcal{F} \wedge CoP2\ \neg^C$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩

CoP3-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow CoP3\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma $\sim TND\ \neg^I \wedge CoP3\ \neg^I$ ⟨*proof*⟩

CoP3-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow CoP3\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma $\sim ECQ\ \neg^C \wedge CoP3\ \neg^C$ ⟨*proof*⟩

XCoP-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow XCoP\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma $\sim TND\ \neg^I \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge XCoP\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩
lemma $\sim TND\ \neg^I \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}3\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge XCoP\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩
lemma $XCoP\ \neg^I \longrightarrow TNDm\ \neg^I$ ⟨*proof*⟩

XCoP-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow XCoP\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma $\sim ECQ\ \neg^C \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge XCoP\ \neg^C$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩
lemma $XCoP\ \neg^C \longrightarrow ECQm\ \neg^C$ ⟨*proof*⟩

7.4 Normality (negative) and its dual (nNor/nDNor)

nNor-I

lemma $nNor\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma $nNor\text{-}I$: $Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow nNor\ \neg^I$ ⟨*proof*⟩

nNor-C

lemma $nNor\text{-}C$: $nNor\ \neg^C$ ⟨*proof*⟩

nDNor-I

lemma $nDNor\text{-}I$: $nDNor\ \neg^I$ ⟨*proof*⟩

nDNor-C

lemma $nDNor\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma $nDNor\text{-}C$: $Fr\text{-}3\ \mathcal{F} \Longrightarrow nDNor\ \neg^C$ ⟨*proof*⟩

7.5 Double negation introduction/elimination (DNI/DNE)

DNI-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow DNI\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma $\sim TNDm\ \neg^I \wedge \mathfrak{F}\ \mathcal{F} \wedge DNI\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩

DNI-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow DNI\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma $\sim ECQ\ \neg^C \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge DNI\ \neg^C$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩

lemma $\sim ECQm\ \neg^C \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}3\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge DNI\ \neg^C$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩
lemma $\sim ECQm\ \neg^C \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}3\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge DNI\ \neg^C$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩

DNE-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow DNE\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma $\sim TND\ \neg^I \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge DNE\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩
lemma $\sim TND\ \neg^I \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}3\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge DNE\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩
lemma $\sim TNDm\ \neg^I \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}3\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge DNE\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩
lemma $\sim TND\ \neg^I \wedge DNE\ \neg^I \wedge DNI\ \neg^I$ ⟨*proof*⟩

DNE-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow DNE\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma $\sim ECQm\ \neg^C \wedge \mathfrak{F}\ \mathcal{F} \wedge DNE\ \neg^C$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩

lemma $\sim ECQ\ \neg^C \wedge DNE\ \neg^C \wedge DNI\ \neg^C$ ⟨*proof*⟩

rDNI-I

lemma $Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow rDNI\ \neg^I$ ⟨*proof*⟩

rDNI-C

lemma $Fr\text{-}3\ \mathcal{F} \Longrightarrow rDNI\ \neg^C$ ⟨*proof*⟩

rDNE-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow rDNE\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma $\sim TNDm\ \neg^I \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge rDNE\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩
lemma $\sim TNDm\ \neg^I \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}3\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge rDNE\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩
lemma $\sim TNDm\ \neg^I \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}3\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge rDNE\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩

rDNE-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow rDNE\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma $\sim ECQm\ \neg^C \wedge \mathfrak{F}\ \mathcal{F} \wedge rDNE\ \neg^C$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩

lemma $\sim ECQm\ \neg^C \wedge rDNE\ \neg^C \wedge rDNI\ \neg^C$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩

7.6 De Morgan laws

DM1/2 (see CoPw)

DM3-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow DM3\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma $\sim TND\ \neg^I \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}2\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge DM3\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩
lemma $\sim TND\ \neg^I \wedge Fr\text{-}1\ \mathcal{F} \wedge Fr\text{-}3\ \mathcal{F} \wedge Fr\text{-}4\ \mathcal{F} \wedge DM3\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩
lemma $\sim TNDm\ \neg^I \wedge DM3\ \neg^I$ ⟨*proof*⟩

DM3-C

lemma $DM3\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma *DM3-C*: $Fr\text{-}1a\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow DM3\ \neg^C$ ⟨*proof*⟩
lemma $\sim ECQm\ \neg^C \wedge \mathfrak{F}\ \mathcal{F} \wedge DM3\ \neg^C$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩

DM4-I

lemma $DM4\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma *DM4-I*: $Fr\text{-}1a\ \mathcal{F} \Longrightarrow DM4\ \neg^I$ ⟨*proof*⟩
lemma $\sim TNDm\ \neg^I \wedge \mathfrak{F}\ \mathcal{F} \wedge DM4\ \neg^I$ **nitpick**[*satisfy,card w=3*] ⟨*proof*⟩

DM4-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{DM4}\ \neg^C$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\sim\!\mathit{ECQ}\ \neg^C \wedge \mathit{Fr\text{-}1}\ \mathcal{F} \wedge \mathit{Fr\text{-}2}\ \mathcal{F} \wedge \mathit{Fr\text{-}4}\ \mathcal{F} \wedge \mathit{DM4}\ \neg^C$ **nitpick**[*satisfy,card w=3*] $\langle \mathit{proof} \rangle$
lemma $\sim\!\mathit{ECQm}\ \neg^C \wedge \mathit{DM4}\ \neg^C$ $\langle \mathit{proof} \rangle$

iDM1/2 (see CoPw)

iDM3-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}inf}\ \mathcal{F} \Longrightarrow \mathit{iDM3}\ \neg^I$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\sim\!\mathit{TND}\ \neg^I \wedge \mathit{Fr\text{-}1}\ \mathcal{F} \wedge \mathit{Fr\text{-}2}\ \mathcal{F} \wedge \mathit{Fr\text{-}4}\ \mathcal{F} \wedge \mathit{iDM3}\ \neg^I$ **nitpick**[*satisfy*] $\langle \mathit{proof} \rangle$
lemma $\sim\!\mathit{TND}\ \neg^I \wedge \mathit{Fr\text{-}1}\ \mathcal{F} \wedge \mathit{Fr\text{-}3}\ \mathcal{F} \wedge \mathit{Fr\text{-}4}\ \mathcal{F} \wedge \mathit{iDM3}\ \neg^I$ **nitpick**[*satisfy*] $\langle \mathit{proof} \rangle$
lemma $\sim\!\mathit{TNDm}\ \neg^I \wedge \mathit{iDM3}\ \neg^I$ $\langle \mathit{proof} \rangle$

iDM3-C

lemma $\mathit{iDM3}\ \neg^C$ **nitpick** $\langle \mathit{proof} \rangle$
lemma *iDM3-C*: $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}inf}\ \mathcal{F} \Longrightarrow \mathit{iDM3}\ \neg^C$ $\langle \mathit{proof} \rangle$

iDM4-I

lemma $\mathit{iDM4}\ \neg^I$ **nitpick** $\langle \mathit{proof} \rangle$
lemma *iDM4-I*: $\mathit{Fr\text{-}inf}\ \mathcal{F} \Longrightarrow \mathit{iDM4}\ \neg^I$ $\langle \mathit{proof} \rangle$

iDM4-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}inf}\ \mathcal{F} \Longrightarrow \mathit{iDM4}\ \neg^C$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\sim\!\mathit{ECQ}\ \neg^C \wedge \mathit{Fr\text{-}1}\ \mathcal{F} \wedge \mathit{Fr\text{-}2}\ \mathcal{F} \wedge \mathit{Fr\text{-}4}\ \mathcal{F} \wedge \mathit{iDM4}\ \neg^C$ **nitpick**[*satisfy*] $\langle \mathit{proof} \rangle$
lemma $\sim\!\mathit{ECQm}\ \neg^C \wedge \mathit{iDM4}\ \neg^C$ $\langle \mathit{proof} \rangle$

7.7 Local contraposition axioms (lCoP)

lCoPw-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{lCoPw}(\rightarrow)\ \neg^I$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\sim\!\mathit{TND}\ \neg^I \wedge \mathit{Fr\text{-}1}\ \mathcal{F} \wedge \mathit{Fr\text{-}2}\ \mathcal{F} \wedge \mathit{Fr\text{-}4}\ \mathcal{F} \wedge \mathit{lCoPw}(\rightarrow)\ \neg^I$ **nitpick**[*satisfy,card w=3*] $\langle \mathit{proof} \rangle$
lemma $\sim\!\mathit{TND}\ \neg^I \wedge \mathit{Fr\text{-}1}\ \mathcal{F} \wedge \mathit{Fr\text{-}3}\ \mathcal{F} \wedge \mathit{Fr\text{-}4}\ \mathcal{F} \wedge \mathit{lCoPw}(\rightarrow)\ \neg^I$ **nitpick**[*satisfy,card w=3*] $\langle \mathit{proof} \rangle$
lemma $\mathit{lCoPw}(\rightarrow)\ \neg^I \longrightarrow \mathit{TNDm}\ \neg^I$ $\langle \mathit{proof} \rangle$

lCoPw-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{lCoPw}(\rightarrow)\ \neg^C$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\sim\!\mathit{ECQ}\ \neg^C \wedge \mathit{Fr\text{-}1}\ \mathcal{F} \wedge \mathit{Fr\text{-}2}\ \mathcal{F} \wedge \mathit{Fr\text{-}4}\ \mathcal{F} \wedge \mathit{lCoPw}(\rightarrow)\ \neg^C$ **nitpick**[*satisfy,card w=3*] $\langle \mathit{proof} \rangle$
lemma $\mathit{lCoPw}(\rightarrow)\ \neg^C \longrightarrow \mathit{ECQm}\ \neg^C$ $\langle \mathit{proof} \rangle$

lCoP1-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{lCoP1}(\rightarrow)\ \neg^I$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\mathit{lCoP1}(\rightarrow)\ \neg^I \longrightarrow \mathit{TND}\ \neg^I$ $\langle \mathit{proof} \rangle$

lCoP1-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{lCoP1}(\rightarrow)\ \neg^C$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\sim\!\mathit{ECQ}\ \neg^C \wedge \mathit{Fr\text{-}1}\ \mathcal{F} \wedge \mathit{Fr\text{-}2}\ \mathcal{F} \wedge \mathit{Fr\text{-}4}\ \mathcal{F} \wedge \mathit{lCoP1}(\rightarrow)\ \neg^C$ **nitpick**[*satisfy,card w=3*] $\langle \mathit{proof} \rangle$
lemma $\mathit{lCoP1}(\rightarrow)\ \neg^C \longrightarrow \mathit{ECQm}\ \neg^C$ $\langle \mathit{proof} \rangle$

lCoP2-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{lCoP2}(\rightarrow)\ \neg^I$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\sim\!\mathit{TND}\ \neg^I \wedge \mathit{Fr\text{-}1}\ \mathcal{F} \wedge \mathit{Fr\text{-}2}\ \mathcal{F} \wedge \mathit{Fr\text{-}4}\ \mathcal{F} \wedge \mathit{lCoP2}(\rightarrow)\ \neg^I$ **nitpick**[*satisfy,card w=3*] $\langle \mathit{proof} \rangle$
lemma $\sim\!\mathit{TND}\ \neg^I \wedge \mathit{Fr\text{-}1}\ \mathcal{F} \wedge \mathit{Fr\text{-}3}\ \mathcal{F} \wedge \mathit{Fr\text{-}4}\ \mathcal{F} \wedge \mathit{lCoP2}(\rightarrow)\ \neg^I$ **nitpick**[*satisfy,card w=3*] $\langle \mathit{proof} \rangle$

lemma $lCoP2(\rightarrow)\ \neg^I \longrightarrow TNDm\ \neg^I$ ⟨*proof*⟩

lCoP2-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow lCoP2(\rightarrow)\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma $lCoP2(\rightarrow)\ \neg^C \longrightarrow ECQ\ \neg^C$ ⟨*proof*⟩

lCoP3-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow lCoP3(\rightarrow)\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma $lCoP3(\rightarrow)\ \neg^I \longrightarrow TND\ \neg^I$ ⟨*proof*⟩

lCoP3-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow lCoP3(\rightarrow)\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma $lCoP3(\rightarrow)\ \neg^C \longrightarrow ECQ\ \neg^C$ ⟨*proof*⟩

7.8 Disjunctive syllogism

DS1-I

lemma $DS1(\rightarrow)\ \neg^I$ ⟨*proof*⟩

DS1-C

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow DS1(\rightarrow)\ \neg^C$ **nitpick** ⟨*proof*⟩
lemma $DS1(\rightarrow)\ \neg^C \longrightarrow ECQ\ \neg^C$ ⟨*proof*⟩

DS2-I

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow DS2(\rightarrow)\ \neg^I$ **nitpick** ⟨*proof*⟩
lemma $DS2(\rightarrow)\ \neg^I \longrightarrow TND\ \neg^I$ ⟨*proof*⟩

DS2-C

lemma $DS2(\rightarrow)\ \neg^C$ ⟨*proof*⟩

end
theory *topo-negation-fixedpoints*
imports *topo-frontier-algebra sse-operation-negative-quantification*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

8 Frontier-based negation - Fixed-points

We define a paracomplete and a paraconsistent negation employing the interior and the closure operation respectively. We explore the use of fixed-point predicates to recover some relevant properties of negation, many of which cannot be readily recovered by just adding semantic conditions. We take the frontier operator as primitive and explore which semantic conditions are minimally required.

definition *neg-I*::$\sigma \Rightarrow \sigma$ $(\neg^I)$ **where** $\neg^I\ \varphi \equiv \mathcal{I}(-\varphi)$
definition *neg-C*::$\sigma \Rightarrow \sigma$ $(\neg^C)$ **where** $\neg^C\ \varphi \equiv \mathcal{C}(-\varphi)$
declare *neg-I-def*[*conn*] *neg-C-def*[*conn*]

Note that all results obtained for fixed-point predicates extend to their associated operators as follows:

lemma $\forall A.\ \gamma^{fp}(A) \land \varphi(A) \preceq \psi(A) \Longrightarrow \forall A.\ (fp\ \gamma)(A) \longrightarrow \varphi(A) \preceq \psi(A)$ ⟨*proof*⟩

lemma $\forall A\ B.\ \gamma^{fp}(A) \wedge \gamma^{fp}(B) \wedge (\varphi\ A\ B) \preceq (\psi\ A\ B) \Longrightarrow \forall A\ B.\ (fp\ \gamma)(A) \wedge (fp\ \gamma)(B) \longrightarrow (\varphi\ A\ B) \preceq (\psi\ A\ B)$ $\langle proof \rangle$

Recall from previous results that if we have Fr(A) then we also have both Cl(A) and Br(A). With this understanding we will tacitly assume the corresponding results for Fr(-) below. Moreover, we obtained countermodels (using Nitpick) for all formulas featuring other combinations (not shown).

8.1 'Explosion' (ECQ) and excluded middle (TND)

TND-I

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A.\ \mathit{Cl}(A) \longrightarrow \mathit{TND}^A\ \neg^I$ $\langle proof \rangle$

ECQ-C

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A.\ \mathit{Op}(A) \longrightarrow \mathit{ECQ}^A\ \neg^C$ $\langle proof \rangle$

8.2 Contraposition rules

CoPw-I

lemma $\forall A\ B.\ \mathit{Br}(-B) \longrightarrow \mathit{CoPw}^{AB}\ \neg^I$ $\langle proof \rangle$
lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Cl}(A) \longrightarrow \mathit{CoPw}^{AB}\ \neg^I$ $\langle proof \rangle$

CoPw-C

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Br}(A) \longrightarrow \mathit{CoPw}^{AB}\ \neg^C$ $\langle proof \rangle$
lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Op}(B) \longrightarrow \mathit{CoPw}^{AB}\ \neg^C$ $\langle proof \rangle$

CoP1-I

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Cl}(A) \longrightarrow \mathit{CoP1}^{AB}\ \neg^I$ $\langle proof \rangle$
lemma $\mathit{Fr\text{-}1b}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Op}(B) \longrightarrow \mathit{CoP1}^{AB}\ \neg^I$ $\langle proof \rangle$
lemma *CoP1-I-rec*: $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}3}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Br}\ (-B) \longrightarrow \mathit{CoP1}^{AB}\ \neg^I$ $\langle proof \rangle$

CoP1-C

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Op}(B) \longrightarrow \mathit{CoP1}^{AB}\ \neg^C$ $\langle proof \rangle$
lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Br}(A) \longrightarrow \mathit{CoP1}^{AB}\ \neg^C$ $\langle proof \rangle$

CoP2-I

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Cl}(A) \longrightarrow \mathit{CoP2}^{AB}\ \neg^I$ $\langle proof \rangle$
lemma $\forall A\ B.\ \mathit{Br}\ (-B) \longrightarrow \mathit{CoP2}^{AB}\ \neg^I$ $\langle proof \rangle$

CoP2-C

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Op}(B) \longrightarrow \mathit{CoP2}^{AB}\ \neg^C$ $\langle proof \rangle$
lemma $\mathit{Fr\text{-}6}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Cl}(A) \longrightarrow \mathit{CoP2}^{AB}\ \neg^C$ $\langle proof \rangle$
lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}3}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Br}(A) \longrightarrow \mathit{CoP2}^{AB}\ \neg^C$ $\langle proof \rangle$

CoP3-I

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Cl}(A) \longrightarrow \mathit{CoP3}^{AB}\ \neg^I$ $\langle proof \rangle$

CoP3-C

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Op}(B) \longrightarrow \mathit{CoP3}^{AB}\ \neg^C$ $\langle proof \rangle$

XCoP-I

lemma $\textit{Fr-2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ Cl(A) \longrightarrow\ \ XCoP^{A\,B}\ \neg^I$ ⟨*proof*⟩
lemma $\forall A\ B.\ Br(-B) \longrightarrow\ \ XCoP^{A\,B}\ \neg^I$ ⟨*proof*⟩

XCoP-C

lemma $\textit{Fr-2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ Op(B) \longrightarrow XCoP^{A\,B}\ \neg^C$ ⟨*proof*⟩
lemma $\textit{Fr-2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \ \forall A\ B.\ Br(A) \longrightarrow XCoP^{A\,B}\ \neg^C$ ⟨*proof*⟩

8.3 Double negation introduction/elimination

DNI-I

lemma $\textit{Fr-1b}\ \mathcal{F} \Longrightarrow \forall A.\ Op(A) \longrightarrow DNI^A\ \neg^I$ ⟨*proof*⟩
lemma $\textit{Fr-2}\ \mathcal{F} \Longrightarrow \textit{Fr-3}\ \mathcal{F} \Longrightarrow \forall A.\ Br\ (-A) \longrightarrow DNI^A\ \neg^I$ ⟨*proof*⟩

DNI-C

lemma $\textit{Fr-2}\ \mathcal{F} \Longrightarrow \forall A.\ Op(A) \longrightarrow DNI^A\ \neg^C$ ⟨*proof*⟩

DNE-I

lemma $\textit{Fr-2}\ \mathcal{F} \Longrightarrow \forall A.\ Cl(A) \longrightarrow DNE^A\ \neg^I$ ⟨*proof*⟩

DNE-C

lemma $\textit{Fr-6}\ \mathcal{F} \Longrightarrow \forall A.\ Cl(A) \longrightarrow DNE^A\ \neg^C$ ⟨*proof*⟩
lemma $\textit{Fr-2}\ \mathcal{F} \Longrightarrow \textit{Fr-3}\ \mathcal{F} \Longrightarrow \forall A.\ Br(A) \longrightarrow DNE^A\ \neg^C$ ⟨*proof*⟩

8.4 De Morgan laws

DM1-I

lemma $\textit{Fr-1b}\ \mathcal{F} \Longrightarrow \forall A\ B.\ DM1^{A\,B}\ \neg^I$ ⟨*proof*⟩
lemma $\textit{Fr-2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ Cl(A) \wedge Cl(B) \longrightarrow DM1^{A\,B}\ \neg^I$ ⟨*proof*⟩

DM1-C

lemma $\textit{Fr-6}\ \mathcal{F} \Longrightarrow \forall A\ B.\ DM1^{A\,B}\ \neg^C$ ⟨*proof*⟩
lemma $\textit{Fr-2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ Br(A) \wedge Br(B) \longrightarrow DM1^{A\,B}\ \neg^C$ ⟨*proof*⟩

DM2-I

lemma $\textit{Fr-1b}\ \mathcal{F} \Longrightarrow \forall A\ B.\ DM2^{A\,B}\ \neg^I$ ⟨*proof*⟩
lemma $\forall A\ B.\ Br(-A) \wedge Br(-B) \longrightarrow DM2^{A\,B}\ \neg^I$ ⟨*proof*⟩

DM2-C

lemma $\textit{Fr-6}\ \mathcal{F} \Longrightarrow \forall A\ B.\ DM2^{A\,B}\ \neg^C$ ⟨*proof*⟩
lemma $\textit{Fr-2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ Op(A) \wedge Op(B) \longrightarrow DM2^{A\,B}\ \neg^C$ ⟨*proof*⟩

DM3-I

lemma $\textit{Fr-2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ Cl(A) \wedge Cl(B) \longrightarrow DM3^{A\,B}\ \neg^I$ ⟨*proof*⟩

DM3-C

lemma $\textit{Fr-1a}\ \mathcal{F} \Longrightarrow \textit{Fr-2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ DM3^{A\,B}\ \neg^C$ ⟨*proof*⟩
lemma $\textit{Fr-2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ Br(A) \wedge Br(B) \longrightarrow DM3^{A\,B}\ \neg^C$ ⟨*proof*⟩

DM4-I

lemma $\textit{Fr-1a}\ \mathcal{F} \Longrightarrow \textit{Fr-2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ DM4^{A\,B}\ \neg^I$ ⟨*proof*⟩
lemma $\forall A\ B.\ Br(-A) \wedge Br(-B) \longrightarrow DM4^{A\,B}\ \neg^I$ ⟨*proof*⟩

DM4-C

lemma $\textit{Fr-2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ Op(A) \wedge Op(B) \longrightarrow DM4^{A\,B}\ \neg^C$ ⟨*proof*⟩
lemma $\textit{Fr-1}\ \mathcal{F} \Longrightarrow \textit{Fr-2}\ \mathcal{F} \Longrightarrow \textit{Fr-4}\ \mathcal{F} \Longrightarrow \forall A\ B.\ Fr(A) \wedge Fr(B) \longrightarrow DM4^{A\,B}\ \neg^C$ ⟨*proof*⟩

8.5 Local contraposition axioms

lCoPw-I

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Cl}(A) \longrightarrow \mathit{lCoPw}^{A\,B}(\rightarrow)\ \neg^{I}$ ⟨*proof*⟩
lemma $\forall A\ B.\ \mathit{Br}(-B) \longrightarrow \mathit{lCoPw}^{A\,B}(\rightarrow)\ \neg^{I}$ ⟨*proof*⟩

lCoPw-C

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Op}(B) \longrightarrow \mathit{lCoPw}^{A\,B}(\rightarrow)\ \neg^{C}$ ⟨*proof*⟩
lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Br}(A) \longrightarrow \mathit{lCoPw}^{A\,B}(\rightarrow)\ \neg^{C}$ ⟨*proof*⟩

lCoP1-I

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Cl}(A) \longrightarrow \mathit{lCoP1}^{A\,B}(\rightarrow)\ \neg^{I}$ ⟨*proof*⟩

lCoP1-C

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Op}(B) \longrightarrow \mathit{lCoP1}^{A\,B}(\rightarrow)\ \neg^{C}$ ⟨*proof*⟩
lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Br}(A) \longrightarrow \mathit{lCoP1}^{A\,B}(\rightarrow)\ \neg^{C}$ ⟨*proof*⟩

lCoP2-I

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Cl}(A) \longrightarrow \mathit{lCoP2}^{A\,B}(\rightarrow)\ \neg^{I}$ ⟨*proof*⟩
lemma $\forall A\ B.\ \mathit{Br}(-B) \longrightarrow \mathit{lCoP2}^{A\,B}(\rightarrow)\ \neg^{I}$ ⟨*proof*⟩

lCoP2-C

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Op}(B) \longrightarrow \mathit{lCoP2}^{A\,B}(\rightarrow)\ \neg^{C}$ ⟨*proof*⟩

lCoP3-I

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Cl}(A) \longrightarrow \mathit{lCoP3}^{A\,B}(\rightarrow)\ \neg^{I}$ ⟨*proof*⟩

lCoP3-C

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Op}(B) \longrightarrow \mathit{lCoP3}^{A\,B}(\rightarrow)\ \neg^{C}$ ⟨*proof*⟩

8.6 Disjunctive syllogism

DS1-I

lemma $\forall A\ B.\ \mathit{DS1}^{A\,B}(\rightarrow)\ \neg^{I}$ ⟨*proof*⟩

DS1-C

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \ \mathit{Op}(A) \longrightarrow \mathit{DS1}^{A\,B}(\rightarrow)\ \neg^{C}$ ⟨*proof*⟩

DS2-I

lemma $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \forall A\ B.\ \mathit{Cl}(A) \longrightarrow \mathit{DS2}^{A\,B}(\rightarrow)\ \neg^{I}$ ⟨*proof*⟩

DS2-C

lemma $\forall A\ B.\ \mathit{DS2}^{A\,B}(\rightarrow)\ \neg^{C}$ ⟨*proof*⟩

end
theory *ex-LFIs*
  imports *topo-negation-conditions*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

9 Example application: Logics of Formal Inconsistency (LFIs)

The LFIs [4] [3] are a family of paraconsistent logics featuring a 'consistency' operator $\circ$ that can be used to recover some classical properties of negation (in particular ECQ). We show how to semantically embed LFIs as extensions of Boolean algebras (here as frontier algebras).

Logical validity can be defined as truth in all worlds/points (i.e. as denoting the top element)

abbreviation *gtrue*::$\sigma \Rightarrow$*bool* ([$\vdash$ -]) **where** $[\vdash\ A] \equiv \forall w.\ A\ w$
lemma *gtrue-def*: $[\vdash A] \equiv A \approx \top$ ⟨*proof*⟩

When defining a logic over an existing algebra we have two choices: a global (truth preserving) and a local (truth-degree preserving) notion of logical consequence. For LFIs we prefer the latter.

abbreviation *conseq-global1*::$\sigma \Rightarrow \sigma \Rightarrow$*bool* ([- $\vdash_g$ -]) **where** $[A \vdash_g B] \equiv [\vdash A] \longrightarrow [\vdash B]$
abbreviation *conseq-global2*::$\sigma \Rightarrow \sigma \Rightarrow \sigma \Rightarrow$*bool* ([-,- $\vdash_g$ -]) **where** $[A_1, A_2 \vdash_g B] \equiv [\vdash A_1] \wedge [\vdash A_2] \longrightarrow [\vdash B]$
abbreviation *conseq-global3*::$\sigma \Rightarrow \sigma \Rightarrow \sigma \Rightarrow \sigma \Rightarrow$*bool* ([-,-,- $\vdash_g$ -]) **where** $[A_1, A_2, A_3 \vdash_g B] \equiv [\vdash A_1] \wedge [\vdash A_2] \wedge [\vdash A_3] \longrightarrow [\vdash B]$
abbreviation *conseq-local1*::$\sigma \Rightarrow \sigma \Rightarrow$*bool* ([- $\vdash$ -]) **where** $[A \vdash B] \equiv A \preceq B$
abbreviation *conseq-local2*::$\sigma \Rightarrow \sigma \Rightarrow \sigma \Rightarrow$*bool* ([-,- $\vdash$ -]) **where** $[A_1, A_2 \vdash B] \equiv A_1 \wedge A_2 \preceq B$
abbreviation *conseq-local3*::$\sigma \Rightarrow \sigma \Rightarrow \sigma \Rightarrow \sigma \Rightarrow$*bool* ([-,-,- $\vdash$ -]) **where** $[A_1, A_2, A_3 \vdash B] \equiv A_1 \wedge A_2 \wedge A_3 \preceq B$

For LFIs we use the (paraconsistent) closure-based negation previously defined (taking frontier as primitive).

abbreviation *cneg*::$\sigma \Rightarrow \sigma$ ($\neg$) **where** $\neg A \equiv \neg^C A$

In terms of the frontier operator the negation looks as follows:

lemma $\neg A \approx -A \vee \mathcal{F}(-A)$ ⟨*proof*⟩
lemma *cneg-prop*: *Fr-2* $\mathcal{F} \Longrightarrow \neg A \approx -A \vee \mathcal{F}(A)$ ⟨*proof*⟩

This negation is of course boldly paraconsistent.

lemma $[a, \neg a \vdash b]$ **nitpick** ⟨*proof*⟩
lemma $[a, \neg a \vdash \neg b]$ **nitpick** ⟨*proof*⟩
lemma $[a, \neg a \vdash_g b]$ **nitpick** ⟨*proof*⟩
lemma $[a, \neg a \vdash_g \neg b]$ **nitpick** ⟨*proof*⟩

We define two pairs of in/consistency operators and show how they relate to each other. Using LFIs terminology, the minimal logic so encoded corresponds to 'RmbC-ciw' (cf. [3]).

abbreviation *op-inc-a*::$\sigma \Rightarrow \sigma$ ($\bullet^A$- [57]58) **where** $\bullet^A A \equiv A \wedge \neg A$
abbreviation *op-con-a*::$\sigma \Rightarrow \sigma$ ($\circ^A$- [57]58) **where** $\circ^A A \equiv -\bullet^A A$
abbreviation *op-inc-b*::$\sigma \Rightarrow \sigma$ ($\bullet^B$- [57]58) **where** $\bullet^B A \equiv \mathcal{B}\ A$
abbreviation *op-con-b*::$\sigma \Rightarrow \sigma$ ($\circ^B$- [57]58) **where** $\circ^B A \equiv \mathcal{B}^c\ A$

Observe that assumming condition Fr-2 are we allowed to exchange A and B variants.

lemma *pincAB*: *Fr-2* $\mathcal{F} \Longrightarrow \bullet^A A \approx \bullet^B A$ ⟨*proof*⟩
lemma *pconAB*: *Fr-2* $\mathcal{F} \Longrightarrow \circ^A A \approx \circ^B A$ ⟨*proof*⟩

Variants A and B give us slightly different properties.

lemma *Prop1*: $\circ^B A \approx \mathcal{I}^{fp}\ A$ ⟨*proof*⟩
lemma $\circ^A A \approx A \rightarrow \mathcal{I}\ A$ **nitpick** ⟨*proof*⟩
lemma *Prop2*: $Cl\ A \longleftrightarrow \circ^A{-A} \approx \top$ ⟨*proof*⟩

lemma $Cl\ A \longrightarrow \circ^B{-}A \approx \top$ **nitpick** $\langle proof \rangle$
lemma *Prop3*: $Cl\ A \longleftrightarrow \bullet^A{-}A \approx \bot$ $\langle proof \rangle$
lemma $Cl\ A \longrightarrow \bullet^B{-}A \approx \bot$ **nitpick** $\langle proof \rangle$
lemma *Prop4*: $Op\ A \longleftrightarrow \circ^B A \approx \top$ $\langle proof \rangle$
lemma $Op\ A \longrightarrow \circ^A A \approx \top$ **nitpick** $\langle proof \rangle$
lemma *Prop5*: $Op\ A \longleftrightarrow \bullet^B A \approx \bot$ $\langle proof \rangle$
lemma $Op\ A \longrightarrow \bullet^A A \approx \bot$ **nitpick** $\langle proof \rangle$

Importantly, LFIs must satisfy the so-called 'principle of gentle explosion'. Only variant A works here:

lemma $[\circ^A a,\ a,\ \neg a \vdash b]$ $\langle proof \rangle$
lemma $[\circ^A a,\ a,\ \neg a \vdash_g b]$ $\langle proof \rangle$
lemma $[\circ^B a,\ a,\ \neg a \vdash b]$ **nitpick** $\langle proof \rangle$
lemma $[\circ^B a,\ a,\ \neg a \vdash_g b]$ **nitpick** $\langle proof \rangle$

In what follows we employ the (minimal) A-variant and verify some properties.

abbreviation *op-inc* :: $\sigma \Rightarrow \sigma$ ($\bullet$- [57]58) **where** $\bullet A \ \equiv \bullet^A A$
abbreviation *op-con* :: $\sigma \Rightarrow \sigma$ ($\circ$- [57]58) **where** $\circ A \ \equiv -\bullet A$

lemma $TND(\neg)$ $\langle proof \rangle$
lemma $ECQm(\neg)$ **nitpick** $\langle proof \rangle$
lemma $Fr\text{-}1b\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow LNC(\neg)$ $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow DNI(\neg)$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow DNE(\neg)$ **nitpick** $\langle proof \rangle$
lemma $Fr\text{-}1b\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow CoPw(\neg)$ $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow CoP1(\neg)$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow CoP2(\neg)$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow CoP3(\neg)$ **nitpick** $\langle proof \rangle$
lemma $Fr\text{-}1a\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow DM3(\neg)$ $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow DM4(\neg)$ **nitpick** $\langle proof \rangle$
lemma $nNor(\neg)$ $\langle proof \rangle$
lemma $Fr\text{-}3\ \mathcal{F} \Longrightarrow nDNor(\neg)$ $\langle proof \rangle$
lemma $Fr\text{-}1b\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow MT0(\neg)$ $\langle proof \rangle$
lemma $Fr\text{-}1b\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow MT1(\neg)$ $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow MT2(\neg)$ **nitpick** $\langle proof \rangle$
lemma $Fr\text{-}1b\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow MT3(\neg)$ $\langle proof \rangle$

We show how all local contraposition variants (lCoP) can be recovered using the consistency operator. Observe that we can recover in the same way other (weaker) properties: CoP, MT and DNI/DNE (local & global).

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow lCoPw(\rightarrow)(\neg)$ **nitpick** $\langle proof \rangle$
lemma *cons-lcop1*: $[\circ b,\ a \rightarrow b \vdash \neg b \rightarrow \neg a]$ $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow lCoP1(\rightarrow)(\neg)$ **nitpick** $\langle proof \rangle$
lemma *cons-lcop2*: $[\circ b,\ a \rightarrow \neg b \vdash b \rightarrow \neg a]$ $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow lCoP2(\rightarrow)(\neg)$ **nitpick** $\langle proof \rangle$
lemma *cons-lcop3*: $[\circ b,\ \neg a \rightarrow b \vdash \neg b \rightarrow a]$ $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow lCoP3(\rightarrow)(\neg)$ **nitpick** $\langle proof \rangle$
lemma *cons-lcop4*: $[\circ b,\ \neg a \rightarrow \neg b \vdash b \rightarrow a]$ $\langle proof \rangle$

Disjunctive syllogism (DS).

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow DS1(\rightarrow)(\neg)$ **nitpick** $\langle proof \rangle$
lemma *cons-ds1*: $[\circ a,\ a \vee b \vdash \neg a \rightarrow b]$ $\langle proof \rangle$
lemma $DS2(\rightarrow)(\neg)$ $\langle proof \rangle$

Further properties.

lemma $[a \wedge \neg a \vdash \neg(\circ a)]$ ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow [\neg(\circ a) \vdash a \wedge \neg a]$ **nitpick** ⟨*proof*⟩
lemma $[\circ a \vdash \neg(a \wedge \neg a)]$ ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow [\neg(a \wedge \neg a) \vdash \circ a]$ **nitpick** ⟨*proof*⟩

The following axioms are commonly employed in the literature on LFIs to obtain stronger logics. We explore under which conditions they can be assumed while keeping the logic boldly paraconsistent.

abbreviation *cf* **where** $cf \equiv DNE(\neg)$
abbreviation *ce* **where** $ce \equiv DNI(\neg)$
abbreviation *ciw* **where** $ciw \equiv \forall P.\ [\vdash \circ P \vee \bullet P]$
abbreviation *ci* **where** $ci \equiv \forall P.\ [\neg(\circ P) \vdash \bullet P]$
abbreviation *cl* **where** $cl \equiv \forall P.\ [\neg(\bullet P) \vdash \circ P]$
abbreviation *ca-conj* **where** $\textit{ca-conj} \equiv \forall P\ Q.\ [\circ P, \circ Q \vdash \circ(P \wedge Q)]$
abbreviation *ca-disj* **where** $\textit{ca-disj} \equiv \forall P\ Q.\ [\circ P, \circ Q \vdash \circ(P \vee Q)]$
abbreviation *ca-impl* **where** $\textit{ca-impl} \equiv \forall P\ Q.\ [\circ P, \circ Q \vdash \circ(P \rightarrow Q)]$
abbreviation *ca* **where** $ca \equiv \textit{ca-conj} \wedge \textit{ca-disj} \wedge \textit{ca-impl}$

cf

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow cf$ **nitpick** ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \wedge cf \wedge \sim ECQm(\neg)$ **nitpick**[*satisfy*] ⟨*proof*⟩

ce

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow ce$ **nitpick** ⟨*proof*⟩
lemma $\textit{Fr-1}\ \mathcal{F} \wedge \textit{Fr-2}\ \mathcal{F} \wedge \textit{Fr-4}\ \mathcal{F} \wedge ce \wedge \sim ECQ(\neg)$ **nitpick**[*satisfy*] ⟨*proof*⟩
lemma $\textit{Fr-1}\ \mathcal{F} \wedge \textit{Fr-3}\ \mathcal{F} \wedge \textit{Fr-4}\ \mathcal{F} \wedge ce \wedge \sim ECQm(\neg)$ **nitpick**[*satisfy*] ⟨*proof*⟩
lemma $\textit{Fr-1a}\ \mathcal{F} \wedge \textit{Fr-2}\ \mathcal{F} \wedge \textit{Fr-3}\ \mathcal{F} \wedge \textit{Fr-4}\ \mathcal{F} \wedge ce \wedge \sim ECQm(\neg)$ **nitpick**[*satisfy*] ⟨*proof*⟩
lemma $\textit{Fr-1b}\ \mathcal{F} \Longrightarrow \textit{Fr-2}\ \mathcal{F} \Longrightarrow ce \longrightarrow ECQm(\neg)$ ⟨*proof*⟩

ciw

lemma *ciw* ⟨*proof*⟩

ci

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow ci$ **nitpick** ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \wedge ci \wedge \sim ECQm(\neg)$ **nitpick**[*satisfy*] ⟨*proof*⟩

cl

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow cl$ **nitpick** ⟨*proof*⟩
lemma $\textit{Fr-1}\ \mathcal{F} \wedge \textit{Fr-3}\ \mathcal{F} \wedge \textit{Fr-4}\ \mathcal{F} \wedge cl \wedge \sim ECQm(\neg)$ **nitpick**[*satisfy*] ⟨*proof*⟩
lemma $\textit{Fr-1a}\ \mathcal{F} \wedge \textit{Fr-2}\ \mathcal{F} \wedge \textit{Fr-3}\ \mathcal{F} \wedge \textit{Fr-4}\ \mathcal{F} \wedge cl \wedge \sim ECQm(\neg)$ **nitpick**[*satisfy*] ⟨*proof*⟩
lemma $\textit{Fr-1b}\ \mathcal{F} \Longrightarrow \textit{Fr-2}\ \mathcal{F} \Longrightarrow cl \longrightarrow ECQ(\neg)$ ⟨*proof*⟩

ca-conj/disj

lemma $\textit{Fr-1a}\ \mathcal{F} \Longrightarrow \textit{Fr-2}\ \mathcal{F} \Longrightarrow \textit{ca-conj}$ ⟨*proof*⟩
lemma $\textit{Fr-1b}\ \mathcal{F} \Longrightarrow \textit{Fr-2}\ \mathcal{F} \Longrightarrow \textit{ca-disj}$ ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \textit{ca-impl}$ **nitpick** ⟨*proof*⟩

ca-impl

lemma $\textit{ca-impl} \wedge \sim ECQ(\neg)$ ⟨*proof*⟩
lemma $\textit{ca-impl} \longrightarrow ECQm(\neg)$ ⟨*proof*⟩

cf & ci

lemma $\textit{Fr-1}\ \mathcal{F} \wedge \textit{Fr-3}\ \mathcal{F} \wedge \textit{Fr-4}\ \mathcal{F} \wedge cf \wedge ci \wedge \sim ECQm(\neg)$ **nitpick**[*satisfy*] ⟨*proof*⟩

lemma *Fr-2* $\mathcal{F}$ ∧ *Fr-3* $\mathcal{F}$ ∧ *Fr-4* $\mathcal{F}$ ∧ *cf* ∧ *ci* ∧ ∼*ECQm*(¬) **nitpick**[*satisfy*] ⟨*proof*⟩
lemma *Fr-1b* $\mathcal{F}$ ∧ *Fr-2* $\mathcal{F}$ ∧ *cf* ∧ *ci* ∧ ∼*ECQ*(¬) ⟨*proof*⟩
lemma $\mathfrak{F}$ $\mathcal{F}$ ∧ *cf* ∧ *ci* ⟶ *ECQm*(¬) ⟨*proof*⟩

cf & cl

lemma *Fr-1* $\mathcal{F}$ ∧ *Fr-3* $\mathcal{F}$ ∧ *Fr-4* $\mathcal{F}$ ∧ *cf* ∧ *cl* ∧ ∼*ECQm*(¬) **nitpick**[*satisfy*] ⟨*proof*⟩
lemma *Fr-2* $\mathcal{F}$ ∧ *Fr-3* $\mathcal{F}$ ∧ *Fr-4* $\mathcal{F}$ ∧ *cf* ∧ *cl* ∧ ∼*ECQm*(¬) **nitpick**[*satisfy*] ⟨*proof*⟩
lemma *Fr-1b* $\mathcal{F}$ ∧ *Fr-2* $\mathcal{F}$ ∧ *cf* ∧ *cl* ⟶ *ECQ*(¬) ⟨*proof*⟩

end
theory *topo-strict-implication*
imports *topo-frontier-algebra*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

10 Strict implication

We can also employ the closure and interior operations to define different sorts of implications. In this section we preliminary explore a sort of strict implication and check some relevant properties.

A 'strict' implication should entail the classical one. Hence we define it using the interior operator.

definition *imp-I*::σ⇒σ⇒σ (**infix** ⇛ *51*) **where** $\alpha \Rrightarrow \beta \equiv \mathcal{I}(\alpha \rightarrow \beta)$
abbreviation *imp-I′*::σ⇒σ⇒σ (**infix** ⇚ *51*) **where** $\beta \Lleftarrow \alpha \equiv \alpha \Rrightarrow \beta$
declare *imp-I-def* [*conn*]

lemma *imp-rel*: $\forall a\ b.\ a \Rrightarrow b \preceq a \rightarrow b$ ⟨*proof*⟩

Under appropriate conditions this implication satisfies a weak variant of the deduction theorem (DT),

lemma *DTw1*: $\forall a\ b.\ a \Rrightarrow b \approx \top \longrightarrow a \preceq b$ ⟨*proof*⟩
lemma *DTw2*: *Fr-2* $\mathcal{F} \Longrightarrow$ *Fr-3* $\mathcal{F} \Longrightarrow \forall a\ b.\ a \preceq b \longrightarrow a \Rrightarrow b \approx \top$ ⟨*proof*⟩

and a variant of modus ponens and modus tollens resp.

lemma *MP*: $\forall a\ b.\ a \wedge (a \Rrightarrow b) \preceq b$ ⟨*proof*⟩
lemma *MT*: $\forall a\ b.\ (a \Rrightarrow b) \wedge -b \preceq -a$ ⟨*proof*⟩

However the full DT (actually right-to-left: implication introduction) is not valid.

lemma *DT1*: $\forall a\ b\ X.\ X \preceq a \Rrightarrow b \longrightarrow X \wedge a \preceq b$ ⟨*proof*⟩
lemma *DT2*: $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall a\ b\ X.\ X \wedge a \preceq b \longrightarrow X \preceq a \Rrightarrow b$ **nitpick** ⟨*proof*⟩

This implication has thus a sort of 'relevant' behaviour. For instance, the following are not valid:

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall a\ b.\ (a \Rrightarrow (b \Rrightarrow a)) \approx \top$ **nitpick** ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall a\ b.\ (a \Rrightarrow ((a \Rrightarrow b) \Rrightarrow b)) \approx \top$ **nitpick** ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ (a \Rrightarrow b) \vee (b \Rrightarrow c) \approx \top$ **nitpick** ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall a\ b.\ ((a \Rrightarrow b) \Rrightarrow a) \Rrightarrow a \approx \top$ **nitpick** ⟨*proof*⟩

In contrast the properties below are valid for appropriate conditions.

lemma *iprop0*: *Fr-2* $\mathcal{F} \Longrightarrow$ *Fr-3* $\mathcal{F} \Longrightarrow \forall a.\ a \Rrightarrow a \approx \top$ ⟨*proof*⟩
lemma *iprop1*: *Fr-2* $\mathcal{F} \Longrightarrow$ *Fr-3* $\mathcal{F} \Longrightarrow \forall a\ b.\ a \wedge (a \Rrightarrow b) \Rrightarrow b \approx \top$ ⟨*proof*⟩

lemma *iprop2*: $Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b.\ a \Rightarrow (b \Rightarrow b) \approx \top$ ⟨*proof*⟩
lemma *iprop3*: $Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b.\ ((a \Rightarrow a) \Rightarrow b) \Rightarrow b \approx \top$ ⟨*proof*⟩
lemma *iprop4*: $Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b.\ (a \wedge b) \Rightarrow a \approx \top$ ⟨*proof*⟩
lemma *iprop5*: $Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b.\ a \Rightarrow (a \vee b) \approx \top$ ⟨*proof*⟩
lemma *iprop6a*: $Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ (a \wedge (b \vee c)) \Rightarrow ((a \wedge b) \vee (a \wedge c)) \approx \top$ ⟨*proof*⟩
lemma *iprop6b*: $Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ (a \wedge (b \vee c)) \Leftarrow ((a \wedge b) \vee (a \wedge c)) \approx \top$ ⟨*proof*⟩

lemma *iprop7′*: $Fr\text{-}1\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ (a \Rightarrow b) \wedge (b \Rightarrow c) \preceq (a \Rightarrow c)$ ⟨*proof*⟩
lemma *iprop7*: $Fr\text{-}1\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ ((a \Rightarrow b) \wedge (b \Rightarrow c)) \Rightarrow (a \Rightarrow c) \approx \top$ ⟨*proof*⟩

lemma *iprop8a′*: $Fr\text{-}1\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ (a \Rightarrow b) \wedge (a \Rightarrow c) \preceq a \Rightarrow (b \wedge c)$ ⟨*proof*⟩
lemma *iprop8b′*: $Fr\text{-}1b\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ (a \Rightarrow b) \wedge (a \Rightarrow c) \succeq a \Rightarrow (b \wedge c)$ ⟨*proof*⟩
lemma *iprop8a*: $Fr\text{-}1\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ ((a \Rightarrow b) \wedge (a \Rightarrow c)) \Rightarrow (a \Rightarrow (b \wedge c)) \approx \top$ ⟨*proof*⟩
lemma *iprop8b*: $Fr\text{-}1b\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ ((a \Rightarrow b) \wedge (a \Rightarrow c)) \Leftarrow (a \Rightarrow (b \wedge c)) \approx \top$ ⟨*proof*⟩

lemma *iprop9a′*: $Fr\text{-}1\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ ((a \Rightarrow b) \wedge (c \Rightarrow b)) \preceq ((a \vee c) \Rightarrow b)$ ⟨*proof*⟩
lemma *iprop9b′*: $Fr\text{-}1b\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ ((a \Rightarrow b) \wedge (c \Rightarrow b)) \succeq ((a \vee c) \Rightarrow b)$ ⟨*proof*⟩
lemma *iprop9a*: $Fr\text{-}1\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ ((a \Rightarrow b) \wedge (c \Rightarrow b)) \Rightarrow ((a \vee c) \Rightarrow b) \approx \top$ ⟨*proof*⟩
lemma *iprop9b*: $Fr\text{-}1b\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ ((a \Rightarrow b) \wedge (c \Rightarrow b)) \Leftarrow ((a \vee c) \Rightarrow b) \approx \top$ ⟨*proof*⟩

lemma *iprop10′*: $Fr\text{-}1\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}4\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ a \Rightarrow (b \Rightarrow c) \preceq (a \Rightarrow b) \Rightarrow (a \Rightarrow c)$ ⟨*proof*⟩
lemma *iprop10*: $Fr\text{-}1\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow Fr\text{-}4\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ (a \Rightarrow (b \Rightarrow c)) \Rightarrow ((a \Rightarrow b) \Rightarrow (a \Rightarrow c)) \approx \top$ ⟨*proof*⟩
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ a \Rightarrow (b \Rightarrow c) \succeq (a \Rightarrow b) \Rightarrow (a \Rightarrow c)$ **nitpick** ⟨*proof*⟩

lemma *iprop11a′*: $Fr\text{-}1\ \mathcal{F} \Longrightarrow \forall a\ b.\ (a \Rightarrow (a \Rightarrow b)) \preceq (a \Rightarrow b)$ ⟨*proof*⟩
lemma *iprop11b′*: $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall a\ b.\ (a \Rightarrow (a \Rightarrow b)) \succeq (a \Rightarrow b)$ ⟨*proof*⟩
lemma *iprop11a*: $Fr\text{-}1\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b.\ (a \Rightarrow (a \Rightarrow b)) \Rightarrow (a \Rightarrow b) \approx \top$ ⟨*proof*⟩
lemma *iprop11b*: $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall a\ b.\ (a \Rightarrow (a \Rightarrow b)) \Leftarrow (a \Rightarrow b) \approx \top$ ⟨*proof*⟩

In particular, 'strenghening the antecedent' is valid only under certain conditions:

lemma *SA′*:$Fr\text{-}1b\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ a \Rightarrow c \preceq (a \wedge b) \Rightarrow c$ ⟨*proof*⟩
lemma *SA*: $Fr\text{-}1b\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ (a \Rightarrow c) \Rightarrow ((a \wedge b) \Rightarrow c) \approx \top$ ⟨*proof*⟩
lemma $Fr\text{-}1a\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow Fr\text{-}4\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ a \Rightarrow c \preceq (a \wedge b) \Rightarrow c$ **nitpick** ⟨*proof*⟩
lemma $Fr\text{-}1a\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow Fr\text{-}4\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ (a \Rightarrow c) \Rightarrow ((a \wedge b) \Rightarrow c) \approx \top$ **nitpick** ⟨*proof*⟩

Similarly, 'weakening the consequent' is valid only under certain conditions.

lemma *WC′*:$Fr\text{-}1b\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ a \Rightarrow c \preceq a \Rightarrow (c \vee b)$ ⟨*proof*⟩
lemma *WC*: $Fr\text{-}1b\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ (a \Rightarrow c) \Rightarrow (a \Rightarrow (c \vee b)) \approx \top$ ⟨*proof*⟩
lemma $Fr\text{-}1a\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow Fr\text{-}4\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ a \Rightarrow c \preceq a \Rightarrow (c \vee b)$ **nitpick** ⟨*proof*⟩
lemma $Fr\text{-}1a\ \mathcal{F} \Longrightarrow Fr\text{-}2\ \mathcal{F} \Longrightarrow Fr\text{-}3\ \mathcal{F} \Longrightarrow Fr\text{-}4\ \mathcal{F} \Longrightarrow \forall a\ b\ c.\ (a \Rightarrow c) \Rightarrow (a \Rightarrow (c \vee b)) \approx \top$ **nitpick** ⟨*proof*⟩

end
theory *ex-subminimal-logics*
imports *topo-negation-conditions topo-strict-implication*

begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

11 Example application: Subintuitionistic and subminimal logics

In this section we examine some special paracomplete logics. The idea is to illustrate an approach by means of which we can obtain logics which are boldly paracomplete and (non-boldly) paraconsistent at the same time, Johansson's 'minimal logic' [7] being the paradigmatic case we aim at modelling.

Drawing upon the literature on Johanson's minimal logic, we introduce an unconstrained propositional constant Q, which we then employ to define a 'rigid' frontier operation $\mathcal{F}'$.

consts $Q::\sigma$
abbreviation $\mathcal{F}' \equiv \lambda X.\ Q$
abbreviation $\mathcal{I}' \equiv \mathcal{I}_F\ \mathcal{F}'$
abbreviation $\mathcal{C}' \equiv \mathcal{C}_F\ \mathcal{F}'$
abbreviation $\mathcal{B}' \equiv \mathcal{B}_F\ \mathcal{F}'$

As defined, $\mathcal{F}'$ (and its corresponding closure operation) satisfies several semantic conditions.

lemma *Fr-1* $\mathcal{F}' \wedge$ *Fr-2* $\mathcal{F}' \wedge$ *Fr-4* $\mathcal{F}'$ $\langle proof \rangle$
lemma *Cl-1* $\mathcal{C}' \wedge$ *Cl-2* $\mathcal{C}' \wedge$ *Cl-4* $\mathcal{C}'$ $\langle proof \rangle$

However Fr-3 is not valid. In fact, adding it by hand would collapse into classical logic (making all sets clopen).

lemma *Fr-3* $\mathcal{F}'$ **nitpick** $\langle proof \rangle$
lemma *Cl-3* $\mathcal{C}'$ **nitpick** $\langle proof \rangle$
lemma *Fr-3* $\mathcal{F}' \Longrightarrow \forall A.\ \mathcal{F}'(A) \approx \bot$ $\langle proof \rangle$

In order to obtain a paracomplete logic not validating ECQ, we define negation as follows,

abbreviation *neg-IC*$::\sigma\Rightarrow\sigma$ $(\neg)$ **where** $\neg A \equiv \mathcal{C}'(\mathcal{I}(-A))$

and observe that some plausible semantic properties obtain:

lemma *Q-def1*: $\forall A.\ Q \approx \neg A \wedge \neg(\neg A)$ $\langle proof \rangle$
lemma *Q-def2*: *Fr-1b* $\mathcal{F} \Longrightarrow \forall A.\ Q \approx \neg(A \vee \neg A)$ $\langle proof \rangle$
lemma *neg-Idef*: $\forall A.\ \neg A \approx \mathcal{I}(-A) \vee Q$ $\langle proof \rangle$
lemma *neg-Cdef*: *Fr-2* $\mathcal{F} \Longrightarrow \forall A.\ \neg A \approx \mathcal{C}(A) \rightarrow Q$ $\langle proof \rangle$

The negation so defined validates some properties corresponding to a (rather weak) paracomplete logic:

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow$ *TND* $\neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow$ *TNDw* $\neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow$ *TNDm* $\neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow$ *ECQ* $\neg$ **nitpick** $\langle proof \rangle$
lemma *ECQw*: *ECQw* $\neg$ $\langle proof \rangle$
lemma *ECQm*: *ECQm* $\neg$ $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow$ *LNC* $\neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow$ *DNI* $\neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow$ *DNE* $\neg$ **nitpick** $\langle proof \rangle$
lemma *CoPw*: *Fr-1b* $\mathcal{F} \Longrightarrow$ *CoPw* $\neg$ $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow$ *CoP1* $\neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow$ *CoP2* $\neg$ **nitpick** $\langle proof \rangle$

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{CoP3}\ \neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{XCoP}\ \neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{DM3}\ \neg$ **nitpick** $\langle proof \rangle$
lemma $\mathit{DM4}$: $\mathit{Fr\text{-}1a}\ \mathcal{F} \Longrightarrow \mathit{DM4}\ \neg\ \langle proof \rangle$
lemma Nor: $\mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}3}\ \mathcal{F} \Longrightarrow \mathit{nNor}\ \neg\ \langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{nDNor}\ \neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{lCoPw}(\rightarrow)\ \neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{lCoP1}(\rightarrow)\ \neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{lCoP2}(\rightarrow)\ \neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{lCoP3}(\rightarrow)\ \neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{DS1}(\rightarrow)\ \neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{DS2}(\rightarrow)\ \neg$ **nitpick** $\langle proof \rangle$

Moreover, we cannot have both DNI and DNE without validating ECQ (thus losing paraconsistency).

lemma $\mathit{DNI}\ \neg \wedge \mathit{DNE}\ \neg \longrightarrow \mathit{ECQ}\ \neg\ \langle proof \rangle$

However, we can have all of De Morgan laws while keeping (non-bold) paraconsistency.

lemma $\sim\!\mathit{ECQ}\ \neg \wedge \mathit{DM1}\ \neg \wedge \mathit{DM2}\ \neg \wedge \mathit{DM3}\ \neg \wedge \mathit{DM4}\ \neg \wedge \mathfrak{F}\ \mathcal{F}$ **nitpick**$[\mathit{satisfy},\mathit{card}\ w{=}3]\ \langle proof \rangle$

Below we combine negation with strict implication and verify some interesting properties. For instance, the following are not valid (and cannot become valid by adding semantic restrictions).

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (\neg a \Rightarrow (a \Rightarrow b)) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (\neg a \rightarrow (a \rightarrow b)) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (a \wedge \neg a \Rightarrow b) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (a \wedge \neg a \rightarrow b) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (a \Rightarrow (b \vee \neg b)) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (a \rightarrow (b \vee \neg b)) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a.\ (a \Rightarrow \neg a) \Rightarrow \neg a \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a.\ (a \rightarrow \neg a) \rightarrow \neg a \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (a \wedge \neg a) \Rightarrow b \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (a \wedge \neg a) \rightarrow b \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ a \Rightarrow (b \vee \neg b) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ a \rightarrow (b \vee \neg b) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (a \leftrightarrow b) \Rightarrow (\neg a \leftrightarrow \neg b) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (a \leftrightarrow b) \rightarrow (\neg a \leftrightarrow \neg b) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (a \Rightarrow b) \wedge (a \Rightarrow \neg b) \Rightarrow \neg a \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (a \rightarrow b) \wedge (a \rightarrow \neg b) \Rightarrow \neg a \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a.\ (\neg a\ \Rightarrow\ \bot) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a.\ (\neg a\ \rightarrow\ \bot) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a.\ (\neg a \Rightarrow \neg(\neg \top)) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a.\ (\neg a \rightarrow \neg(\neg \top)) \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a.\ \neg(\neg(\neg a)) \Rightarrow \neg a \approx \top$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a.\ \neg(\neg(\neg a)) \rightarrow \neg a \approx \top$ **nitpick** $\langle proof \rangle$

The (weak) local contraposition axiom is indeed valid under appropriate conditions.

lemma lCoPw: $\mathit{Fr\text{-}1}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}3}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}4}\ \mathcal{F} \Longrightarrow \mathit{lCoPw}(\Rightarrow)\ \neg\ \langle proof \rangle$
lemma $\mathit{lCoPw\text{-}strict}$: $\mathfrak{F}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (a \Rightarrow b) \Rightarrow (\neg b \Rightarrow \neg a) \approx \top\ \langle proof \rangle$

However, other (local) contraposition axioms are not valid.

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{lCoP1}(\Rightarrow)\ \neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{lCoP2}(\Rightarrow)\ \neg$ **nitpick** $\langle proof \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{lCoP3}(\Rightarrow)\ \neg$ **nitpick** $\langle proof \rangle$

And this time no variant of disjunctive syllogism is valid.

lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{DS1}(\Rightarrow)\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{DS2}(\Rightarrow)\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{DS2}(\Rightarrow)\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\mathfrak{F}\ \mathcal{F} \Longrightarrow \mathit{DS4}(\Rightarrow)\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$

Interestingly, one of the local contraposition axioms (lCoP1) follows from DNI.

lemma $\mathit{DNI\text{-}lCoP1}$: $\mathit{Fr\text{-}1}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}3}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}4}\ \mathcal{F} \Longrightarrow \mathit{DNI}\ \neg \longrightarrow \mathit{lCoP1}(\Rightarrow)\ \neg\ \langle \mathit{proof} \rangle$

This entails some other interesting results.

lemma $\mathit{DNI\text{-}CoP1}$: $\mathit{Fr\text{-}1b}\ \mathcal{F} \Longrightarrow \mathit{DNI}\ \neg \Longrightarrow \mathit{CoP1}\ \neg\ \langle \mathit{proof} \rangle$
lemma $\mathit{CoP1\text{-}LNC}$: $\mathit{CoP1}\ \neg \Longrightarrow \mathit{LNC}\ \neg\ \langle \mathit{proof} \rangle$
lemma $\mathit{DNI\text{-}LNC}$: $\mathit{Fr\text{-}1b}\ \mathcal{F} \Longrightarrow \mathit{DNI}\ \neg \Longrightarrow \mathit{LNC}\ \neg\ \langle \mathit{proof} \rangle$

The following variants of modus tollens also obtain.

lemma MT: $\mathit{Fr\text{-}1}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}3}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ (a \Rightarrow b) \wedge \neg b \preceq \neg a\ \langle \mathit{proof} \rangle$
lemma MT': $\mathit{Fr\text{-}1}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}2}\ \mathcal{F} \Longrightarrow \mathit{Fr\text{-}3}\ \mathcal{F} \Longrightarrow \forall\, a\ b.\ ((a \Rightarrow b) \wedge \neg b) \Rightarrow \neg a \approx \top\ \langle \mathit{proof} \rangle$

We now semantically characterize (an approximation of) Johansson's Minimal Logic along with some exemplary 'subminimal' logics (observing that many more are possible). We check some relevant properties.

abbreviation $\mathit{JML} \equiv \mathfrak{F}\ \mathcal{F} \wedge \mathit{DNI}\ \neg$
abbreviation $\mathit{SML1} \equiv \mathfrak{F}\ \mathcal{F}$
abbreviation $\mathit{SML2} \equiv \mathit{Fr\text{-}1}\ \mathcal{F} \wedge \mathit{Fr\text{-}2}\ \mathcal{F} \wedge \mathit{Fr\text{-}3}\ \mathcal{F}$
abbreviation $\mathit{SML3} \equiv \mathit{Fr\text{-}1}\ \mathcal{F}$
abbreviation $\mathit{SML4} \equiv \mathit{Fr\text{-}1b}\ \mathcal{F}$

TND:

lemma $\mathit{JML} \Longrightarrow \mathit{TND}\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\mathit{JML} \Longrightarrow \mathit{TNDw}\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\mathit{JML} \Longrightarrow \mathit{TNDm}\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$

ECQ:

lemma $\mathit{JML} \Longrightarrow \mathit{ECQ}\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\mathit{ECQw}\ \neg\ \langle \mathit{proof} \rangle$
lemma $\mathit{ECQm}\ \neg\ \langle \mathit{proof} \rangle$

LNC:

lemma $\mathit{JML} \Longrightarrow \mathit{LNC}\ \neg\ \langle \mathit{proof} \rangle$
lemma $\mathit{SML1} \Longrightarrow \mathit{LNC}\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$

(r)DNI/DNE:

lemma $\mathit{JML} \Longrightarrow \mathit{DNI}\ \neg\ \langle \mathit{proof} \rangle$
lemma $\mathit{SML1} \Longrightarrow \mathit{rDNI}\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\mathit{JML} \Longrightarrow \mathit{rDNE}\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$

CoP/MT:

lemma $\mathit{SML4} \Longrightarrow \mathit{CoPw}\ \neg\ \langle \mathit{proof} \rangle$
lemma $\mathit{JML} \Longrightarrow \mathit{CoP1}\ \neg\ \langle \mathit{proof} \rangle$
lemma $\mathit{SML1} \Longrightarrow \mathit{MT1}\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\mathit{JML} \Longrightarrow \mathit{MT2}\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$
lemma $\mathit{JML} \Longrightarrow \mathit{MT3}\ \neg$ **nitpick** $\langle \mathit{proof} \rangle$

XCoP:

lemma *JML* $\Longrightarrow$ *XCoP* ¬ **nitpick** ⟨*proof*⟩

DM3/4:

lemma *JML* $\Longrightarrow$ *DM3* ¬ **nitpick** ⟨*proof*⟩
lemma *SML3* $\Longrightarrow$ *DM4* ¬ ⟨*proof*⟩
lemma *SML4* $\Longrightarrow$ *DM4* ¬ **nitpick** ⟨*proof*⟩

nNor/nDNor:

lemma *SML2* $\Longrightarrow$ *nNor* ¬ ⟨*proof*⟩
lemma *SML3* $\Longrightarrow$ *nNor* ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *nDNor* ¬ **nitpick** ⟨*proof*⟩

lCoP classical:

lemma *JML* $\Longrightarrow$ *lCoPw*($\rightarrow$) ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *lCoP1*($\rightarrow$) ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *lCoP2*($\rightarrow$) ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *lCoP3*($\rightarrow$) ¬ **nitpick** ⟨*proof*⟩

DS classical:

lemma *JML* $\Longrightarrow$ *DS1*($\rightarrow$) ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *DS2*($\rightarrow$) ¬ **nitpick** ⟨*proof*⟩

lCoP strict:

lemma *SML1* $\Longrightarrow$ *lCoPw*($\Rightarrow$) ¬ ⟨*proof*⟩
lemma *SML2* $\Longrightarrow$ *lCoPw*($\Rightarrow$) ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *lCoP1*($\Rightarrow$) ¬ ⟨*proof*⟩
lemma *SML1* $\Longrightarrow$ *lCoP1*($\Rightarrow$) ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *lCoP2*($\Rightarrow$) ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *lCoP3*($\Rightarrow$) ¬ **nitpick** ⟨*proof*⟩

lMT strict:

lemma *SML2* $\Longrightarrow$ *lMT0*($\Rightarrow$) ¬ ⟨*proof*⟩
lemma *SML3* $\Longrightarrow$ *lMT0*($\Rightarrow$) ¬ ⟨*proof*⟩
lemma *SML4* $\Longrightarrow$ *lMT0*($\Rightarrow$) ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *lMT1*($\Rightarrow$) ¬ ⟨*proof*⟩
lemma *SML1* $\Longrightarrow$ *lMT1*($\Rightarrow$) ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *lMT2*($\Rightarrow$) ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *lMT3*($\Rightarrow$) ¬ **nitpick** ⟨*proof*⟩

DS strict:

lemma *JML* $\Longrightarrow$ *DS1*($\Rightarrow$) ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *DS2*($\Rightarrow$) ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *DS3*($\Rightarrow$) ¬ **nitpick** ⟨*proof*⟩
lemma *JML* $\Longrightarrow$ *DS4*($\Rightarrow$) ¬ **nitpick** ⟨*proof*⟩

end
theory *topo-derivative-algebra*
imports *topo-operators-derivative*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

12 Derivative algebra

The closure of a set A ($\mathcal{C}(A)$) can be seen as the set A augmented by (i) its boundary points, or (ii) its accumulation/limit points. We explore the second variant by drawing on the notion of derivative algebra.

Declares a primitive (unconstrained) derivative (aka. derived-set) operation and defines others from it.

consts $\mathcal{D}::\sigma\Rightarrow\sigma$
abbreviation $\mathcal{I} \equiv \mathcal{I}_D\ \mathcal{D}$ — interior
abbreviation $\mathcal{C} \equiv \mathcal{C}_D\ \mathcal{D}$ — closure
abbreviation $\mathcal{B} \equiv \mathcal{B}_D\ \mathcal{D}$ — border
abbreviation $\mathcal{F} \equiv \mathcal{F}_D\ \mathcal{D}$ — frontier
abbreviation $\mathcal{K} \equiv \mathcal{K}_D\ \mathcal{D}$ — coherence

12.1 Basic properties

Verifies minimal conditions under which operators resulting from conversion functions coincide.

lemma *ICdual*: $\mathcal{I} \equiv \mathcal{C}^d$ ⟨*proof*⟩
lemma *ICdual′*: $\mathcal{C} \equiv \mathcal{I}^d$ ⟨*proof*⟩
lemma *BI-rel*: $\mathcal{B} \equiv \mathcal{B}_I\ \mathcal{I}$ ⟨*proof*⟩
lemma *IB-rel*: $\mathcal{I} \equiv \mathcal{I}_B\ \mathcal{B}$ ⟨*proof*⟩
lemma *BC-rel*: $\mathcal{B} \equiv \mathcal{B}_C\ \mathcal{C}$ ⟨*proof*⟩
lemma *CB-rel*: $\mathcal{C} \equiv \mathcal{C}_B\ \mathcal{B}$ ⟨*proof*⟩
lemma *FI-rel*: $\mathcal{F} \equiv \mathcal{F}_I\ \mathcal{I}$ ⟨*proof*⟩
lemma *FC-rel*: $\mathcal{F} \equiv \mathcal{F}_C\ \mathcal{C}$ ⟨*proof*⟩
lemma *FB-rel*: $\mathcal{F} \equiv \mathcal{F}_B\ \mathcal{B}$ ⟨*proof*⟩

Recall that derivative and coherence operations cannot be obtained from either interior, closure, border nor frontier. The derivative operation can indeed be seen as being more fundamental than the other ones.

Fixed-point and other operators are interestingly related.

lemma *fp1*: $\mathcal{I}^{fp} \equiv \mathcal{B}^c$ ⟨*proof*⟩
lemma *fp2*: $\mathcal{B}^{fp} \equiv \mathcal{I}^c$ ⟨*proof*⟩
lemma *fp3*: $\mathcal{C}^{fp} \equiv \mathcal{B}^d$ ⟨*proof*⟩
lemma *fp4*: $(\mathcal{B}^d)^{fp} \equiv \mathcal{C}$ ⟨*proof*⟩
lemma *fp5*: $\mathcal{F}^{fp} \equiv \mathcal{B} \sqcup (\mathcal{C}^c)$ ⟨*proof*⟩
lemma *fp6*: $\mathcal{D}^{fp} \equiv \mathcal{K} \sqcup (\mathcal{C}^c)$ ⟨*proof*⟩

Different inter-relations (some redundant ones are kept to help the provers).

lemma *monI*: *Der-1b* $\mathcal{D} \Longrightarrow$ *MONO* $\mathcal{I}$ ⟨*proof*⟩
lemma *monC*: *Der-1b* $\mathcal{D} \Longrightarrow$ *MONO* $\mathcal{C}$ ⟨*proof*⟩
lemma *pB1*: $\forall A.\ \mathcal{B}\ A \approx A \leftharpoondown \mathcal{I}\ A$ ⟨*proof*⟩
lemma *pB2*: $\forall A.\ \mathcal{B}\ A \approx A \wedge \mathcal{F}\ A$ ⟨*proof*⟩
lemma *pB3*: $\forall A.\ \mathcal{B}(-A) \approx -A \wedge \mathcal{F}\ A$ ⟨*proof*⟩
lemma *pB4*: $\forall A.\ \mathcal{B}(-A) \approx -A \wedge \mathcal{C}\ A$ ⟨*proof*⟩
lemma *pB5*: *Der-1b* $\mathcal{D} \Longrightarrow \forall A.\ \mathcal{B}(\mathcal{C}\ A) \preceq (\mathcal{B}\ A) \vee \mathcal{B}(-A)$ ⟨*proof*⟩
lemma *pF1*: $\forall A.\ \mathcal{F}\ A \approx \mathcal{C}\ A \leftharpoondown \mathcal{I}\ A$ ⟨*proof*⟩
lemma *pF2*: $\forall A.\ \mathcal{F}\ A \approx \mathcal{C}\ A \wedge \mathcal{C}(-A)$ ⟨*proof*⟩
lemma *pF3*: $\forall A.\ \mathcal{F}\ A \approx \mathcal{B}\ A \vee \mathcal{B}(-A)$ ⟨*proof*⟩
lemma *pF4*: *Der-1* $\mathcal{D} \Longrightarrow$ *Der-4e* $\mathcal{D} \Longrightarrow \forall A.\ \mathcal{F}(\mathcal{I}\ A) \preceq \mathcal{F}\ A$ ⟨*proof*⟩
lemma *pF5*: *Der-1* $\mathcal{D} \Longrightarrow$ *Der-4e* $\mathcal{D} \Longrightarrow \forall A.\ \mathcal{F}(\mathcal{C}\ A) \preceq \mathcal{F}\ A$ ⟨*proof*⟩
lemma *pA1*: $\forall A.\ A \approx \mathcal{I}\ A \vee \mathcal{B}\ A$ ⟨*proof*⟩

lemma *pA2*: $\forall A.\ A \approx \mathcal{C}\ A \leftharpoondown \mathcal{B}(-A)$ ⟨*proof*⟩
lemma *pC1*: $\forall A.\ \mathcal{C}\ A \approx A \vee \mathcal{B}(-A)$ ⟨*proof*⟩
lemma *pC2*: $\forall A.\ \mathcal{C}\ A \approx A \vee \mathcal{F}\ A$ ⟨*proof*⟩
lemma *pI1*: $\forall A.\ \mathcal{I}\ A \approx A \leftharpoondown \mathcal{B}\ A$ ⟨*proof*⟩
lemma *pI2*: $\forall A.\ \mathcal{I}\ A \approx A \leftharpoondown \mathcal{F}\ A$ ⟨*proof*⟩

lemma *IC-imp*: *Der-1* $\mathcal{D} \Longrightarrow$ *Der-3* $\mathcal{D} \Longrightarrow \forall A\ B.\ \mathcal{I}(A \rightarrow B) \preceq \mathcal{C}\ A \rightarrow \mathcal{C}\ B$ ⟨*proof*⟩

Define some fixed-point predicates and prove some properties.

abbreviation *openset* (*Op*) **where** *Op* $A \equiv$ *fp* $\mathcal{I}\ A$
abbreviation *closedset* (*Cl*) **where** *Cl* $A \equiv$ *fp* $\mathcal{C}\ A$
abbreviation *borderset* (*Br*) **where** *Br* $A \equiv$ *fp* $\mathcal{B}\ A$
abbreviation *frontierset* (*Fr*) **where** *Fr* $A \equiv$ *fp* $\mathcal{F}\ A$

lemma *Int-Open*: *Der-1a* $\mathcal{D} \Longrightarrow$ *Der-4e* $\mathcal{D} \Longrightarrow \forall A.$ *Op*$(\mathcal{I}\ A)$ ⟨*proof*⟩
lemma *Cl-Closed*: *Der-1a* $\mathcal{D} \Longrightarrow$ *Der-4e* $\mathcal{D} \Longrightarrow \forall A.$ *Cl*$(\mathcal{C}\ A)$ ⟨*proof*⟩
lemma *Br-Border*: *Der-1b* $\mathcal{D} \Longrightarrow \forall A.$ *Br*$(\mathcal{B}\ A)$ ⟨*proof*⟩

In contrast, there is no analogous fixed-point result for frontier:

lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow \forall A.$ *Fr*$(\mathcal{F}\ A)$ **nitpick** ⟨*proof*⟩

lemma *OpCldual*: $\forall A.$ *Cl* $A \longleftrightarrow$ *Op*$(-A)$ ⟨*proof*⟩
lemma *ClOpdual*: $\forall A.$ *Op* $A \longleftrightarrow$ *Cl*$(-A)$ ⟨*proof*⟩
lemma *Fr-ClBr*: $\forall A.$ *Fr*$(A) = ($*Cl*$(A) \wedge$ *Br*$(A))$ ⟨*proof*⟩
lemma *Cl-F*: *Der-1* $\mathcal{D} \Longrightarrow$ *Der-4e* $\mathcal{D} \Longrightarrow \forall A.$ *Cl*$(\mathcal{F}\ A)$ ⟨*proof*⟩

12.2 Further properties

The definitions and theorems below are well known in the literature (e.g. [9]). Here we uncover the minimal conditions under which they hold (taking derivative operation as primitive).

lemma *Cl-Bzero*: $\forall A.$ *Cl* $A \longleftrightarrow \mathcal{B}(-A) \approx \bot$ ⟨*proof*⟩
lemma *Op-Bzero*: $\forall A.$ *Op* $A \longleftrightarrow\ \mathcal{B}\ A \approx \bot$ ⟨*proof*⟩
lemma *Br-boundary*: $\forall A.$ *Br*$(A) \longleftrightarrow \mathcal{I}\ A \approx \bot$ ⟨*proof*⟩
lemma *Fr-nowhereDense*: $\forall A.$ *Fr*$(A) \longrightarrow \mathcal{I}(\mathcal{C}\ A) \approx \bot$ ⟨*proof*⟩
lemma *Cl-FB*: $\forall A.$ *Cl* $A \longleftrightarrow \mathcal{F}\ A \approx \mathcal{B}\ A$ ⟨*proof*⟩
lemma *Op-FB*: $\forall A.$ *Op* $A \longleftrightarrow \mathcal{F}\ A \approx \mathcal{B}(-A)$ ⟨*proof*⟩
lemma *Clopen-Fzero*: $\forall A.$ *Cl* $A \wedge$ *Op* $A \longleftrightarrow \mathcal{F}\ A \approx \bot$ ⟨*proof*⟩

lemma *Int-sup-closed*: *Der-1b* $\mathcal{D} \Longrightarrow$ *supremum-closed* $(\lambda A.$ *Op* $A)$ ⟨*proof*⟩
lemma *Int-meet-closed*: *Der-1a* $\mathcal{D} \Longrightarrow$ *meet-closed* $(\lambda A.$ *Op* $A)$ ⟨*proof*⟩
lemma *Int-inf-closed*: *Der-inf* $\mathcal{D} \Longrightarrow$ *infimum-closed* $(\lambda A.$ *Op* $A)$ ⟨*proof*⟩
lemma *Cl-inf-closed*: *Der-1b* $\mathcal{D} \Longrightarrow$ *infimum-closed* $(\lambda A.$ *Cl* $A)$ ⟨*proof*⟩
lemma *Cl-join-closed*: *Der-1a* $\mathcal{D} \Longrightarrow$ *join-closed* $(\lambda A.$ *Cl* $A)$ ⟨*proof*⟩
lemma *Cl-sup-closed*: *Der-inf* $\mathcal{D} \Longrightarrow$ *supremum-closed* $(\lambda A.$ *Cl* $A)$ ⟨*proof*⟩
lemma *Br-inf-closed*: *Der-1b* $\mathcal{D} \Longrightarrow$ *infimum-closed* $(\lambda A.$ *Br* $A)$ ⟨*proof*⟩
lemma *Fr-inf-closed*: *Der-1b* $\mathcal{D} \Longrightarrow$ *infimum-closed* $(\lambda A.$ *Fr* $A)$ ⟨*proof*⟩
lemma *Br-Fr-join*: *Der-1* $\mathcal{D} \Longrightarrow$ *Der-4e* $\mathcal{D} \Longrightarrow \forall A\ B.$ *Br* $A \wedge$ *Fr* $B \longrightarrow$ *Br*$(A \vee B)$ ⟨*proof*⟩
lemma *Fr-join-closed*: *Der-1* $\mathcal{D} \Longrightarrow$ *Der-4e* $\mathcal{D} \Longrightarrow$ *join-closed* $(\lambda A.$ *Fr* $A)$ ⟨*proof*⟩

Introduces a predicate for indicating that two sets are disjoint and proves some properties.

abbreviation *Disj* $A\ B \equiv A \wedge B \approx \bot$

lemma *Disj-comm*: $\forall A\ B.$ *Disj* $A\ B \longrightarrow$ *Disj* $B\ A$ ⟨*proof*⟩
lemma *Disj-IF*: $\forall A.$ *Disj* $(\mathcal{I}\ A)\ (\mathcal{F}\ A)$ ⟨*proof*⟩

lemma *Disj-B*: $\forall A.\ Disj\ (\mathcal{B}\ A)\ (\mathcal{B}(-A))$ ⟨*proof*⟩
lemma *Disj-I*: $\forall A.\ Disj\ (\mathcal{I}\ A)\ (-A)$ ⟨*proof*⟩
lemma *Disj-BCI*: $\forall A.\ Disj\ (\mathcal{B}(\mathcal{C}\ A))\ (\mathcal{I}(-A))$ ⟨*proof*⟩
lemma *Disj-CBI*: $Der\text{-}1b\ \mathcal{D} \Longrightarrow Der\text{-}4e\ \mathcal{D} \Longrightarrow \forall A.\ Disj\ (\mathcal{C}(\mathcal{B}(-A)))\ (\mathcal{I}(-A))$ ⟨*proof*⟩

Introduce a predicate for indicating that two sets are separated and proves some properties.

definition $Sep\ A\ B \equiv Disj\ (\mathcal{C}\ A)\ B \wedge Disj\ (\mathcal{C}\ B)\ A$

lemma *Sep-comm*: $\forall A\ B.\ Sep\ A\ B \longrightarrow Sep\ B\ A$ ⟨*proof*⟩
lemma *Sep-disj*: $\forall A\ B.\ Sep\ A\ B \longrightarrow Disj\ A\ B$ ⟨*proof*⟩
lemma *Sep-I*: $Der\text{-}1\ \mathcal{D} \Longrightarrow Der\text{-}4e\ \mathcal{D} \Longrightarrow \forall A.\ Sep\ (\mathcal{I}\ A)\ (\mathcal{I}\ (-A))$ ⟨*proof*⟩

lemma *Sep-sub*: $Der\text{-}1b\ \mathcal{D} \Longrightarrow \forall A\ B\ C\ D.\ Sep\ A\ B \wedge C \preceq A \wedge D \preceq B \longrightarrow Sep\ C\ D$ ⟨*proof*⟩
lemma *Sep-Cl-diff*: $Der\text{-}1b\ \mathcal{D} \Longrightarrow \forall A\ B.\ Cl(A) \wedge Cl(B) \longrightarrow Sep\ (A \leftharpoondown B)\ (B \leftharpoondown A)$ ⟨*proof*⟩
lemma *Sep-Op-diff*: $Der\text{-}1b\ \mathcal{D} \Longrightarrow \forall A\ B.\ Op(A) \wedge Op(B) \longrightarrow Sep\ (A \leftharpoondown B)\ (B \leftharpoondown A)$ ⟨*proof*⟩
lemma *Sep-Cl*: $\forall A\ B.\ Cl(A) \wedge Cl(B) \wedge Disj\ A\ B \longrightarrow Sep\ A\ B$ ⟨*proof*⟩
lemma *Sep-Op*: $Der\text{-}1b\ \mathcal{D} \Longrightarrow \forall A\ B.\ Op(A) \wedge Op(B) \wedge Disj\ A\ B \longrightarrow Sep\ A\ B$ ⟨*proof*⟩
lemma $Der\text{-}1a\ \mathcal{D} \Longrightarrow \forall A\ B\ C.\ Sep\ A\ B \wedge Sep\ A\ C \longrightarrow Sep\ A\ (B \vee C)$ ⟨*proof*⟩

Verifies a neighborhood-based definition of interior.

definition $nbhd\ A\ p \equiv \exists E.\ E \preceq A \wedge Op(E) \wedge (E\ p)$
lemma *nbhd-def2*: $Der\text{-}1\ \mathcal{D} \Longrightarrow Der\text{-}4e\ \mathcal{D} \Longrightarrow \forall A\ p.\ (nbhd\ A\ p) = (\mathcal{I}\ A\ p)$ ⟨*proof*⟩

lemma *I-def′-lr′*: $\forall A\ p.\ (\mathcal{I}\ A\ p) \longrightarrow (\exists E.\ (\mathcal{I}\ E\ p) \wedge E \preceq A)$ ⟨*proof*⟩
lemma *I-def′-rl′*: $Der\text{-}1b\ \mathcal{D} \Longrightarrow \forall A\ p.\ (\mathcal{I}\ A\ p) \longleftarrow (\exists E.\ (\mathcal{I}\ E\ p) \wedge E \preceq A)$ ⟨*proof*⟩
lemma *I-def′*: $Der\text{-}1b\ \mathcal{D} \Longrightarrow \forall A\ p.\ (\mathcal{I}\ A\ p) \longleftrightarrow (\exists E.\ (\mathcal{I}\ E\ p) \wedge E \preceq A)$ ⟨*proof*⟩

Explore the Barcan and converse Barcan formulas.

lemma *Barcan-I*: $Der\text{-}inf\ \mathcal{D} \Longrightarrow \forall P.\ (\forall x.\ \mathcal{I}(P\ x)) \preceq \mathcal{I}(\forall x.\ P\ x)$ ⟨*proof*⟩
lemma *Barcan-C*: $Der\text{-}inf\ \mathcal{D} \Longrightarrow \forall P.\ \mathcal{C}(\exists x.\ P\ x) \preceq (\exists x.\ \mathcal{C}(P\ x))$ ⟨*proof*⟩
lemma *CBarcan-I*: $Der\text{-}1b\ \mathcal{D} \Longrightarrow \forall P.\ \mathcal{I}(\forall x.\ P\ x) \preceq (\forall x.\ \mathcal{I}(P\ x))$ ⟨*proof*⟩
lemma *CBarcan-C*: $Der\text{-}1b\ \mathcal{D} \Longrightarrow \forall P.\ (\exists x.\ \mathcal{C}(P\ x)) \preceq \mathcal{C}(\exists x.\ P\ x)$ ⟨*proof*⟩

end
theory *ex-LFUs*
  imports *topo-derivative-algebra sse-operation-negative*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

13 Example application: Logics of Formal Undeterminedness (LFUs)

The LFUs [10] [4] are a family of paracomplete logics featuring a 'determinedness' operator ∘ that can be used to recover some classical properties of negation (in particular TND). LFUs behave in a sense dually to LFIs. Both can be semantically embedded as extensions of Boolean algebras. Here we show how to semantically embed LFUs as derivative algebras.

(We rename (classical) meta-logical negation to avoid terminological confusion)

abbreviation *cneg*::*bool*⇒*bool* (∼- [*40*]*40*) **where** $\sim\varphi \equiv \neg\varphi$

Logical validity can be defined as truth in all worlds/points (i.e. as denoting the top element)

abbreviation *gtrue*::σ⇒*bool* ([⊢ -]) **where** $[\vdash\ A] \equiv \forall w.\ A\ w$

lemma *gtrue-def*: $[\vdash A] \equiv A \approx \top$ ⟨*proof*⟩

As for LFIs, we focus on the local (truth-degree preserving) notion of logical consequence.

abbreviation *conseq-local1*::$\sigma \Rightarrow \sigma \Rightarrow bool$ ([- ⊢ -]) **where** $[A \vdash B] \equiv A \preceq B$
abbreviation *conseq-local2*::$\sigma \Rightarrow \sigma \Rightarrow \sigma \Rightarrow bool$ ([-,- ⊢ -]) **where** $[A_1, A_2 \vdash B] \equiv A_1 \wedge A_2 \preceq B$
abbreviation *conseq-local12*::$\sigma \Rightarrow \sigma \Rightarrow \sigma \Rightarrow bool$ ([- ⊢ -,-]) **where** $[A \vdash B_1, B_2] \equiv A \preceq B_1 \vee B_2$

For LFUs we use the interior-based negation previously defined (taking derivative as primitive).

definition *ineg*::$\sigma \Rightarrow \sigma$ ($\neg$) **where** $\neg A \equiv \mathcal{I}(-A)$
declare *ineg-def*[*conn*]

In terms of the derivative operator the negation looks as follows:

lemma *ineg-prop*: $\neg A \approx -(\mathcal{D}\ A) \leftharpoondown A$ ⟨*proof*⟩

This negation is of course paracomplete.

lemma $[\vdash a \vee \neg a]$ **nitpick** ⟨*proof*⟩

We define two pairs of in/determinedness operators and show how they relate to each other.

abbreviation *op-det*::$\sigma \Rightarrow \sigma$ (∘- [*57*]*58*) **where** $\circ A \equiv \mathcal{B}^d\ A$
abbreviation *op-ind*::$\sigma \Rightarrow \sigma$ (•- [*57*]*58*) **where** $\bullet A \equiv -\circ A$

lemma *op-det-def*: $\circ a \approx a \vee \neg a$ ⟨*proof*⟩
lemma *Prop1*: $\circ A \approx \mathcal{C}^{fp}\ A$ ⟨*proof*⟩
lemma *Prop2*: $Op\ A \longleftrightarrow \circ -A \approx \top$ ⟨*proof*⟩
lemma *Prop3*: $Op\ A \longleftrightarrow \bullet -A \approx \bot$ ⟨*proof*⟩
lemma *Prop4*: $Cl\ A \longleftrightarrow \circ A \approx \top$ ⟨*proof*⟩
lemma *Prop5*: $Cl\ A \longleftrightarrow \bullet A \approx \bot$ ⟨*proof*⟩

Analogously as for LFIs, LFUs provide means for recovering the principle of excluded middle.

lemma $[\Gamma \vdash \bullet a, a \vee \neg a]$ ⟨*proof*⟩
lemma $[\Gamma, \circ a \vdash a \vee \neg a]$ ⟨*proof*⟩

lemma *TNDm*($\neg$) **nitpick** ⟨*proof*⟩
lemma *ECQ*($\neg$) ⟨*proof*⟩
lemma *Der-3* $\mathcal{D} \Longrightarrow$ *LNC*($\neg$) ⟨*proof*⟩
lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow$ *DNI*($\neg$) **nitpick** ⟨*proof*⟩
lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow$ *DNE*($\neg$) **nitpick** ⟨*proof*⟩
lemma *Der-1b* $\mathcal{D} \Longrightarrow$ *CoPw*($\neg$) ⟨*proof*⟩
lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow$ *CoP1*($\neg$) **nitpick** ⟨*proof*⟩
lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow$ *CoP2*($\neg$) **nitpick** ⟨*proof*⟩
lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow$ *CoP3*($\neg$) **nitpick** ⟨*proof*⟩
lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow$ *DM3*($\neg$) **nitpick** ⟨*proof*⟩
lemma *Der-1a* $\mathcal{D} \Longrightarrow$ *DM4*($\neg$) ⟨*proof*⟩
lemma *Der-3* $\mathcal{D} \Longrightarrow$ *nNor*($\neg$) ⟨*proof*⟩
lemma *nDNor*($\neg$) ⟨*proof*⟩
lemma *Der-1b* $\mathcal{D} \Longrightarrow$ *MT0*($\neg$) ⟨*proof*⟩
lemma *Der-1b* $\mathcal{D} \Longrightarrow$ *Der-3* $\mathcal{D} \Longrightarrow$ *MT1*($\neg$) ⟨*proof*⟩
lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow$ *MT2*($\neg$) **nitpick** ⟨*proof*⟩
lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow$ *MT3*($\neg$) **nitpick** ⟨*proof*⟩

We show how all local contraposition variants (lCoP) can be recovered using the determinedness operator. Observe that we can recover in the same way other (weaker) properties: CoP, MT and DNI/DNE (local & global).

lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow$ *lCoPw*($\rightarrow$)($\neg$) **nitpick** ⟨*proof*⟩

lemma *det-lcop1*: $[\circ a,\ a \rightarrow b \vdash \neg b \rightarrow \neg a]$ ⟨*proof*⟩
lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow$ *lCoP1*$(\rightarrow)(\neg)$ **nitpick** ⟨*proof*⟩
lemma *det-lcop2*: $[\circ a,\ a \rightarrow \neg b \vdash b \rightarrow \neg a]$ ⟨*proof*⟩
lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow$ *lCoP2*$(\rightarrow)(\neg)$ **nitpick** ⟨*proof*⟩
lemma *det-lcop3*: $[\circ a,\ \neg a \rightarrow b \vdash \neg b \rightarrow a]$ ⟨*proof*⟩
lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow$ *lCoP3*$(\rightarrow)(\neg)$ **nitpick** ⟨*proof*⟩
lemma *det-lcop4*: $[\circ a,\ \neg a \rightarrow \neg b \vdash b \rightarrow a]$ ⟨*proof*⟩

Disjunctive syllogism (DS).

lemma *DS1*$(\rightarrow)(\neg)$ ⟨*proof*⟩
lemma $\mathfrak{D}\ \mathcal{D} \Longrightarrow$ *DS2*$(\rightarrow)(\neg)$ **nitpick** ⟨*proof*⟩
lemma *det-ds2*: $[\circ a,\ \neg a \rightarrow b \vdash a \vee b]$ ⟨*proof*⟩

end
theory *topo-border-algebra*
imports *topo-operators-basic*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

14 Border algebra

We define a border algebra in an analogous fashion to the well-known closure/interior algebras. We also verify a few interesting properties.

Declares a primitive (unconstrained) border operation and defines others from it.

consts $\mathcal{B}{::}\sigma \Rightarrow \sigma$
abbreviation $\mathcal{I} \equiv \mathcal{I}_B\ \mathcal{B}$ — interior
abbreviation $\mathcal{C} \equiv \mathcal{C}_B\ \mathcal{B}$ — closure
abbreviation $\mathcal{F} \equiv \mathcal{F}_B\ \mathcal{B}$ — frontier

14.1 Basic properties

Verifies minimal conditions under which operators resulting from conversion functions coincide.

lemma *ICdual*: $\mathcal{I} \equiv \mathcal{C}^d$ ⟨*proof*⟩
lemma *ICdual'*: $\mathcal{C} \equiv \mathcal{I}^d$ ⟨*proof*⟩
lemma *FI-rel*: *Br-1* $\mathcal{B} \Longrightarrow \mathcal{F} \equiv \mathcal{F}_I\ \mathcal{I}$ ⟨*proof*⟩
lemma *IF-rel*: *Br-1* $\mathcal{B} \Longrightarrow \mathcal{I} \equiv \mathcal{I}_F\ \mathcal{F}$ ⟨*proof*⟩
lemma *FC-rel*: *Br-1* $\mathcal{B} \Longrightarrow \mathcal{F} \equiv \mathcal{F}_C\ \mathcal{C}$ ⟨*proof*⟩
lemma *CF-rel*: *Br-1* $\mathcal{B} \Longrightarrow \mathcal{C} \equiv \mathcal{C}_F\ \mathcal{F}$ ⟨*proof*⟩
lemma *BI-rel*: *Br-1* $\mathcal{B} \Longrightarrow \mathcal{B} \equiv \mathcal{B}_I\ \mathcal{I}$ ⟨*proof*⟩
lemma *BC-rel*: *Br-1* $\mathcal{B} \Longrightarrow \mathcal{B} \equiv \mathcal{B}_C\ \mathcal{C}$ ⟨*proof*⟩
lemma *BF-rel*: *Br-1* $\mathcal{B} \Longrightarrow \mathcal{B} \equiv \mathcal{B}_F\ \mathcal{F}$ ⟨*proof*⟩

Fixed-point and other operators are interestingly related.

lemma *fp1*: *Br-1* $\mathcal{B} \Longrightarrow \mathcal{I}^{fp} \equiv \mathcal{B}^c$ ⟨*proof*⟩
lemma *fp2*: *Br-1* $\mathcal{B} \Longrightarrow \mathcal{B}^{fp} \equiv \mathcal{I}^c$ ⟨*proof*⟩
lemma *fp3*: *Br-1* $\mathcal{B} \Longrightarrow \mathcal{C}^{fp} \equiv \mathcal{B}^d$ ⟨*proof*⟩
lemma *fp4*: *Br-1* $\mathcal{B} \Longrightarrow (\mathcal{B}^d)^{fp} \equiv \mathcal{C}$ ⟨*proof*⟩
lemma *fp5*: *Br-1* $\mathcal{B} \Longrightarrow \mathcal{F}^{fp} \equiv \mathcal{B} \sqcup (\mathcal{C}^c)$ ⟨*proof*⟩

Define some fixed-point predicates and prove some properties.

abbreviation *openset* (*Op*) **where** *Op* $A \equiv$ *fp* $\mathcal{I}\ A$
abbreviation *closedset* (*Cl*) **where** *Cl* $A \equiv$ *fp* $\mathcal{C}\ A$

abbreviation *borderset* (*Br*) **where** $Br\ A \equiv fp\ \mathcal{B}\ A$
abbreviation *frontierset* (*Fr*) **where** $Fr\ A \equiv fp\ \mathcal{F}\ A$

lemma *Int-Open*: $Br\text{-}1\ \mathcal{B} \Longrightarrow Br\text{-}3\ \mathcal{B} \Longrightarrow \forall A.\ Op(\mathcal{I}\ A)$ ⟨*proof*⟩
lemma *Cl-Closed*: $Br\text{-}1\ \mathcal{B} \Longrightarrow Br\text{-}3\ \mathcal{B} \Longrightarrow \forall A.\ Cl(\mathcal{C}\ A)$ ⟨*proof*⟩
lemma *Br-Border*: $Br\text{-}1\ \mathcal{B} \Longrightarrow \forall A.\ Br(\mathcal{B}\ A)$ ⟨*proof*⟩

In contrast, there is no analogous fixed-point result for frontier:

lemma $\mathfrak{B}\ \mathcal{B} \Longrightarrow \forall A.\ Fr(\mathcal{F}\ A)$ **nitpick** ⟨*proof*⟩

lemma *OpCldual*: $\forall A.\ Cl\ A \longleftrightarrow Op(-A)$ ⟨*proof*⟩
lemma *ClOpdual*: $\forall A.\ Op\ A \longleftrightarrow Cl(-A)$ ⟨*proof*⟩
lemma *Fr-ClBr*: $Br\text{-}1\ \mathcal{B} \Longrightarrow \forall A.\ Fr(A) = (Cl(A) \wedge Br(A))$ ⟨*proof*⟩
lemma *Cl-F*: $Br\text{-}1\ \mathcal{B} \Longrightarrow Br\text{-}3\ \mathcal{B} \Longrightarrow \forall A.\ Cl(\mathcal{F}\ A)$ ⟨*proof*⟩

end
theory *topo-closure-algebra*
  imports *topo-operators-basic*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

15 Closure algebra

We define a topological Boolean algebra with a primitive closure operator and verify a few properties.

Declares a primitive (unconstrained) closure operation and defines others from it.

consts $\mathcal{C}{::}\sigma{\Rightarrow}\sigma$
abbreviation $\mathcal{I} \equiv \mathcal{C}^d$ — interior
abbreviation $\mathcal{B} \equiv \mathcal{B}_C\ \mathcal{C}$ — border
abbreviation $\mathcal{F} \equiv \mathcal{F}_C\ \mathcal{C}$ — frontier

15.1 Basic properties

Verifies minimal conditions under which operators resulting from conversion functions coincide.

lemma *ICdual'*: $\mathcal{C} \equiv \mathcal{I}^d$ ⟨*proof*⟩
lemma *IB-rel*: $Cl\text{-}2\ \mathcal{C} \Longrightarrow \mathcal{I} \equiv \mathcal{I}_B\ \mathcal{B}$ ⟨*proof*⟩
lemma *IF-rel*: $Cl\text{-}2\ \mathcal{C} \Longrightarrow \mathcal{I} \equiv \mathcal{I}_F\ \mathcal{F}$ ⟨*proof*⟩
lemma *CB-rel*: $Cl\text{-}2\ \mathcal{C} \Longrightarrow \mathcal{C} \equiv \mathcal{C}_B\ \mathcal{B}$ ⟨*proof*⟩
lemma *CF-rel*: $Cl\text{-}2\ \mathcal{C} \Longrightarrow \mathcal{C} \equiv \mathcal{C}_F\ \mathcal{F}$ ⟨*proof*⟩
lemma *BI-rel*: $\mathcal{B} \equiv \mathcal{B}_I\ \mathcal{I}$ ⟨*proof*⟩
lemma *BF-rel*: $Cl\text{-}2\ \mathcal{C} \Longrightarrow \mathcal{B} \equiv \mathcal{B}_F\ \mathcal{F}$ ⟨*proof*⟩
lemma *FI-rel*: $\mathcal{F} \equiv \mathcal{F}_I\ \mathcal{I}$ ⟨*proof*⟩
lemma *FB-rel*: $Cl\text{-}2\ \mathcal{C} \Longrightarrow \mathcal{F} \equiv \mathcal{F}_B\ \mathcal{B}$ ⟨*proof*⟩

Fixed-point and other operators are interestingly related.

lemma *fp1*: $Cl\text{-}2\ \mathcal{C} \Longrightarrow \mathcal{I}^{fp} \equiv \mathcal{B}^c$ ⟨*proof*⟩
lemma *fp2*: $Cl\text{-}2\ \mathcal{C} \Longrightarrow \mathcal{B}^{fp} \equiv \mathcal{I}^c$ ⟨*proof*⟩
lemma *fp3*: $Cl\text{-}2\ \mathcal{C} \Longrightarrow \mathcal{C}^{fp} \equiv \mathcal{B}^d$ ⟨*proof*⟩
lemma *fp4*: $Cl\text{-}2\ \mathcal{C} \Longrightarrow (\mathcal{B}^d)^{fp} \equiv \mathcal{C}$ ⟨*proof*⟩
lemma *fp5*: $Cl\text{-}2\ \mathcal{C} \Longrightarrow \mathcal{F}^{fp} \equiv \mathcal{B} \sqcup (\mathcal{C}^c)$ ⟨*proof*⟩

Define some fixed-point predicates and prove some properties.

abbreviation *openset* (*Op*) **where** $Op\ A \equiv fp\ \mathcal{I}\ A$
abbreviation *closedset* (*Cl*) **where** $Cl\ A \equiv fp\ \mathcal{C}\ A$
abbreviation *borderset* (*Br*) **where** $Br\ A \equiv fp\ \mathcal{B}\ A$
abbreviation *frontierset* (*Fr*) **where** $Fr\ A \equiv fp\ \mathcal{F}\ A$

lemma *Int-Open*: $Cl\text{-}4\ \mathcal{C} \Longrightarrow \forall A.\ Op(\mathcal{I}\ A)$ ⟨*proof*⟩
lemma *Cl-Closed*: $Cl\text{-}4\ \mathcal{C} \Longrightarrow \forall A.\ Cl(\mathcal{C}\ A)$ ⟨*proof*⟩
lemma *Br-Border*: $Cl\text{-}1b\ \mathcal{C} \Longrightarrow \forall A.\ Br(\mathcal{B}\ A)$ ⟨*proof*⟩

In contrast, there is no analogous fixed-point result for frontier:

lemma $\mathfrak{C}\ \mathcal{C} \Longrightarrow \forall A.\ Fr(\mathcal{F}\ A)$ **nitpick** ⟨*proof*⟩

lemma *OpCldual*: $\forall A.\ Cl\ A \longleftrightarrow Op(-A)$ ⟨*proof*⟩
lemma *ClOpdual*: $\forall A.\ Op\ A \longleftrightarrow Cl(-A)$ ⟨*proof*⟩
lemma *Fr-ClBr*: $Cl\text{-}2\ \mathcal{C} \Longrightarrow \forall A.\ Fr(A) = (Cl(A) \wedge Br(A))$ ⟨*proof*⟩
lemma *Cl-F*: $Cl\text{-}1b\ \mathcal{C} \Longrightarrow Cl\text{-}2\ \mathcal{C} \Longrightarrow Cl\text{-}4\ \mathcal{C} \Longrightarrow \forall A.\ Cl(\mathcal{F}\ A)$ ⟨*proof*⟩

end
theory *topo-interior-algebra*
imports *topo-operators-basic*
begin
nitpick-params[*assms=true*, *user-axioms=true*, *show-all*, *expect=genuine*, *format=3*]

16 Interior algebra

We define a topological Boolean algebra taking the interior operator as primitive and verify some properties.

Declares a primitive (unconstrained) interior operation and defines others from it.

consts $\mathcal{I}::\sigma\Rightarrow\sigma$
abbreviation $\mathcal{C} \equiv \mathcal{I}^d$ — closure
abbreviation $\mathcal{B} \equiv \mathcal{B}_I\ \mathcal{I}$ — border
abbreviation $\mathcal{F} \equiv \mathcal{F}_I\ \mathcal{I}$ — frontier

16.1 Basic properties

Verifies minimal conditions under which operators resulting from conversion functions coincide.

lemma *ICdual*: $\mathcal{I} \equiv \mathcal{C}^d$ ⟨*proof*⟩
lemma *IB-rel*: $Int\text{-}2\ \mathcal{I} \Longrightarrow \mathcal{I} \equiv \mathcal{I}_B\ \mathcal{B}$ ⟨*proof*⟩
lemma *IF-rel*: $Int\text{-}2\ \mathcal{I} \Longrightarrow \mathcal{I} \equiv \mathcal{I}_F\ \mathcal{F}$ ⟨*proof*⟩
lemma *CB-rel*: $Int\text{-}2\ \mathcal{I} \Longrightarrow \mathcal{C} \equiv \mathcal{C}_B\ \mathcal{B}$ ⟨*proof*⟩
lemma *CF-rel*: $Int\text{-}2\ \mathcal{I} \Longrightarrow \mathcal{C} \equiv \mathcal{C}_F\ \mathcal{F}$ ⟨*proof*⟩
lemma *BC-rel*: $\mathcal{B} \equiv \mathcal{B}_C\ \mathcal{C}$ ⟨*proof*⟩
lemma *BF-rel*: $Int\text{-}2\ \mathcal{I} \Longrightarrow \mathcal{B} \equiv \mathcal{B}_F\ \mathcal{F}$ ⟨*proof*⟩
lemma *FC-rel*: $\mathcal{F} \equiv \mathcal{F}_C\ \mathcal{C}$ ⟨*proof*⟩
lemma *FB-rel*: $Int\text{-}2\ \mathcal{I} \Longrightarrow \mathcal{F} \equiv \mathcal{F}_B\ \mathcal{B}$ ⟨*proof*⟩

Fixed-point and other operators are interestingly related.

lemma *fp1*: $Int\text{-}2\ \mathcal{I} \Longrightarrow \mathcal{I}^{fp} \equiv \mathcal{B}^c$ ⟨*proof*⟩
lemma *fp2*: $Int\text{-}2\ \mathcal{I} \Longrightarrow \mathcal{B}^{fp} \equiv \mathcal{I}^c$ ⟨*proof*⟩
lemma *fp3*: $Int\text{-}2\ \mathcal{I} \Longrightarrow \mathcal{C}^{fp} \equiv \mathcal{B}^d$ ⟨*proof*⟩
lemma *fp4*: $Int\text{-}2\ \mathcal{I} \Longrightarrow (\mathcal{B}^d)^{fp} \equiv \mathcal{C}$ ⟨*proof*⟩
lemma *fp5*: $Int\text{-}2\ \mathcal{I} \Longrightarrow \mathcal{F}^{fp} \equiv \mathcal{B} \sqcup (\mathcal{C}^c)$ ⟨*proof*⟩

Define some fixed-point predicates and prove some properties.

abbreviation *openset* (*Op*) **where** $Op\ A \equiv fp\ \mathcal{I}\ A$
abbreviation *closedset* (*Cl*) **where** $Cl\ A \equiv fp\ \mathcal{C}\ A$
abbreviation *borderset* (*Br*) **where** $Br\ A \equiv fp\ \mathcal{B}\ A$
abbreviation *frontierset* (*Fr*) **where** $Fr\ A \equiv fp\ \mathcal{F}\ A$

lemma *Int-Open*: $\textit{Int-4}\ \mathcal{I} \Longrightarrow \forall A.\ Op(\mathcal{I}\ A)$ ⟨*proof*⟩
lemma *Cl-Closed*: $\textit{Int-4}\ \mathcal{I} \Longrightarrow \forall A.\ Cl(\mathcal{C}\ A)$ ⟨*proof*⟩
lemma *Br-Border*: $\textit{Int-1a}\ \mathcal{I} \Longrightarrow \forall A.\ Br(\mathcal{B}\ A)$ ⟨*proof*⟩

In contrast, there is no analogous fixed-point result for frontier:

lemma $\mathfrak{I}\ \mathcal{I} \Longrightarrow \forall A.\ Fr(\mathcal{F}\ A)$ **nitpick** ⟨*proof*⟩

lemma *OpCldual*: $\forall A.\ Cl\ A \longleftrightarrow Op(-A)$ ⟨*proof*⟩
lemma *ClOpdual*: $\forall A.\ Op\ A \longleftrightarrow Cl(-A)$ ⟨*proof*⟩
lemma *Fr-ClBr*: $\textit{Int-2}\ \mathcal{I} \Longrightarrow \forall A.\ Fr(A) = (Cl(A) \wedge Br(A))$ ⟨*proof*⟩
lemma *Cl-F*: $\textit{Int-1a}\ \mathcal{I} \Longrightarrow \textit{Int-2}\ \mathcal{I} \Longrightarrow \textit{Int-4}\ \mathcal{I} \Longrightarrow \forall A.\ Cl(\mathcal{F}\ A)$ ⟨*proof*⟩

end

References

[1] C. Benzmüller. Universal (meta-)logical reasoning: Recent successes. *Science of Computer Programming*, 172:48–62, 2019. Preprint: http://doi.org/10.13140/RG.2.2.11039.61609/2.

[2] C. Benzmüller and L. Paulson. Quantified multimodal logics in simple type theory. *Logica Universalis (Special Issue on Multimodal Logics)*, 7(1):7–20, 2013. Preprint: http://christoph-benzmueller.de/papers/J23.pdf.

[3] W. Carnielli, M. E. Coniglio, and D. Fuenmayor. Logics of formal inconsistency enriched with replacement: an algebraic and modal account. *arXiv*, math.LO(2003.09522), 2020.

[4] W. Carnielli, M. E. Coniglio, and J. Marcos. Logics of formal inconsistency. In *Handbook of philosophical logic*, pages 1–93. Springer, 2007.

[5] L. Esakia. Intuitionistic logic and modality via topology. *Annals of Pure and Applied Logic*, 127(1-3):155–170, 2004.

[6] F. Hausdorff. *Grundzüge der Mengenlehre*, volume 7. von Veit, 1914.

[7] I. Johansson. Der Minimalkalkül, ein reduzierter intuitionistischer Formalismus. *Compositio mathematica*, 4:119–136, 1937.

[8] K. Kuratowski. Sur l'opération $\bar{\text{A}}$ de l'analysis situs. *Fundamenta Mathematicae*, 3(1):182–199, 1922.

[9] K. Kuratowski. *Topology: Volume I*, volume 1. Elsevier, 2014.

[10] J. Marcos. Nearly every normal modal logic is paranormal. *Logique et Analyse*, 48(189/192):279–300, 2005.

[11] J. C. C. McKinsey and A. Tarski. The algebra of topology. *Annals of mathematics*, pages 141–191, 1944.

[12] M. Zarycki. Quelques notions fondamentales de l'analysis situs au point de vue de l'algèbre de la logique. *Fundamenta Mathematicae*, 9(1):3–15, 1927. English translation by Mark Bowron: https://www.researchgate.net/scientific-contributions/Miron-Zarycki-2016157096.

[13] M. Zarycki. Allgemeine Eigenschaften der cantorschen Kohärenzen. *Transactions of the American Mathematical Society*, 30(3):498–506, 1928. English translation by Mark Bowron: https://www.researchgate.net/scientific-contributions/Miron-Zarycki-2016157096.

[14] M. Zarycki. Some properties of the derived set operation in abstract spaces. *Nauk. Zap. Ser. Fiz.-Mat.*, 5:22–33, 1947. English translation by Mark Bowron: https://www.researchgate.net/scientific-contributions/Miron-Zarycki-2016157096.