Modal quantales, involutive Quantales, Dedekind Quantales

Cameron Calk and Georg Struth

September 13, 2023

Abstract

This AFP entry provides mathematical components for modal quantales, involutive quantales and Dedekind quantales. Modal quantales are simple extensions of modal Kleene algebras useful for the verification of recursive programs. Involutive quantales appear in the study of C^*-algebras. Dedekind quantales are relatives of Tarski's relation algebras, hence relevant to program verification and beyond that to higher rewriting. We also provide components for weaker variants such as Kleene algebras with converse and modal Kleene algebras with converse.

Contents

1 Introductory Remarks

In this AFP entry we provide mathematical components for modal quantales, involutive quantales and Dedekind quantales. Modal quantales are simple extensions of modal Kleene algebras that can be used in the verification of recursive programs [6]. Involutive quantales appear in the study of C^*-algebras [8]. Dedekind quantales, categorifications of which are known as *modular quantaloids* [9], are relatives of Tarski's relation algebras [11], and hence relevant to program verification as well. We also provide components for weaker variants such as Kleene algebras and modal Kleene algebras with converse.

Our main interest in these structures comes from recent applications in higher-dimensional rewriting [2, 3], where they are used in coherence proofs for rewriting systems based on computads or polygraphs. This includes proofs of coherent Church-Rosser theorems and coherent Newman's lemmas. A more long-term programme considers the formalisation of algebraic aspects of higher rewriting with proof assistants.

Modal quantales have previously been studied in [4], where it is shown, for instance, that any category can be lifted to a modal quantale at powerset level. Such lifting results will be formalised in a companion AFP entry.

Dedekind quantales give also rise to intuitionistic modal algebras, as the results in this AFP entry show. In particular, the set of all subidentities or coreflexives of a Dedekind quantale forms a complete Heyting algebra (aka frame or locale), on which modal box and diamond operators can be defined. A paper explaining these results is in preparation [7]. A further application of Dedekind quantales lies once again in higher-dimensional rewriting [2, 3]. Any groupoid, in particular, can be lifted to a Dedekind quantale at powerset level, a result which will once again be formalised in a companion AFP entry.

Our components build on extant AFP components for Kleene algebras [1], modal Kleene algebras [5] and quantales [10].

Georg Struth is grateful for an invited professorship at École polytechnique and a fellowship at the Collegium de Lyon, Institute of Advanced Study, during which most of this formalisation work has been done.

2 Modal Kleene algebra based on domain and range semirings

theory *Modal-Kleene-Algebra-Var*
  imports *KAD.Domain-Semiring KAD.Range-Semiring*

begin

notation *domain-op* (*dom*)
notation *range-op* (*cod*)

subclass (**in** *domain-semiring*) *dioid-one-zero*⟨*proof*⟩

subclass (**in** *range-semiring*) *dioid-one-zero*
  ⟨*proof*⟩

2.1 Modal semirings

The following modal semirings are based on domain and range semirings instead of antidomain and antirange semirings, as in the AFP entry for Kleene algebra with domain.

class *dr-modal-semiring* = *domain-semiring* + *range-semiring* +
  assumes *dc-compat* [*simp*]: *dom* (*cod x*) = *cod x*
  and *cd-compat* [*simp*]: *cod* (*dom x*) = *dom x*

begin

sublocale *msrdual*: *dr-modal-semiring* (+) $\lambda x\ y.\ y \cdot x$ *1 0 cod* (≤) (<) *dom*
  ⟨*proof*⟩

lemma *d-cod-fix*: (*dom x* = *x*) = (*x* = *cod x*)
  ⟨*proof*⟩

lemma *local-var*: ($x \cdot y$ = *0*) = (*cod x* · *dom y* = *0*)
  ⟨*proof*⟩

lemma *fbdia-conjugation*: (*fd x* (*dom p*) · *dom q* = *0*) = (*dom p* · *bd x* (*dom q*) = *0*)
  ⟨*proof*⟩

end

2.2 Modal Kleene algebra

class *dr-modal-kleene-algebra* = *dr-modal-semiring* + *kleene-algebra*

end

3 Kleene algebra with converse

theory *Kleene-Algebra-Converse*
imports *Kleene-Algebra.Kleene-Algebra*

begin

We start from involutive dioids and Kleene algebra and then add a so-called strong Gelfand property to obtain an operation of converse that is closer to algebras of paths and relations.

3.1 Involutive Kleene algebra

class *invol-op* =
fixes *invol* :: $'a \Rightarrow 'a$ (-° [*101*] *100*)

class *involutive-dioid* = *dioid-one-zero* + *invol-op* +
assumes *inv-invol* [*simp*]: $(x^\circ)^\circ = x$
and *inv-contrav* [*simp*]: $(x \cdot y)^\circ = y^\circ \cdot x^\circ$
and *inv-sup* [*simp*]: $(x + y)^\circ = x^\circ + y^\circ$

begin

lemma *inv-zero* [*simp*]: $0^\circ = 0$
⟨*proof*⟩

lemma *inv-one* [*simp*]: $1^\circ = 1$
⟨*proof*⟩

lemma *inv-iso*: $x \le y \Longrightarrow x^\circ \le y^\circ$
⟨*proof*⟩

lemma *inv-adj*: $(x^\circ \le y) = (x \le y^\circ)$
⟨*proof*⟩

end

Here is an equivalent axiomatisation from Doornbos, Backhouse and van der Woude's paper on a calculational approach to mathematical induction.

class *involutive-dioid-alt* = *dioid-one-zero* +
fixes *inv-alt* :: $'a \Rightarrow 'a$
assumes *inv-alt*: (*inv-alt* $x \le y$) = ($x \le$ *inv-alt* y)
and *inv-alt-contrav* [*simp*]: *inv-alt* $(x \cdot y)$ = *inv-alt* $y \cdot$ *inv-alt* x

begin

lemma *inv-alt-invol* [*simp*]: *inv-alt* (*inv-alt* x) = x
⟨*proof*⟩

lemma *inv-alt-add*: *inv-alt* $(x + y) =$ *inv-alt* x + *inv-alt* y
⟨*proof*⟩

sublocale *altinv*: *involutive-dioid* - - - - - - *inv-alt*
⟨*proof*⟩

end

sublocale *involutive-dioid* ⊆ *altinv*: *involutive-dioid-alt* - - - - - - *invol*
⟨*proof*⟩

class *involutive-kleene-algebra* = *involutive-dioid* + *kleene-algebra*

begin

lemma *inv-star*: $(x^\star)^\circ = (x^\circ)^\star$
⟨*proof*⟩

end

3.2 Kleene algebra with converse

The name "strong Gelfand property" has been borrowed from Palmigiano and Re.

class *dioid-converse* = *involutive-dioid* +
assumes *strong-gelfand*: $x \leq x \cdot x^\circ \cdot x$

lemma (**in** *dioid-converse*) *subid-conv*: $x \leq 1 \Longrightarrow x^\circ = x$
⟨*proof*⟩

class *kleene-algebra-converse* = *involutive-kleene-algebra* + *dioid-converse*

end

4 Modal Kleene algebra with converse

theory *Modal-Kleene-Algebra-Converse*
imports *Modal-Kleene-Algebra-Var Kleene-Algebra-Converse*

begin

Here we mainly study the interaction of converse with domain and codomain.

4.1 Involutive modal Kleene algebras

class *involutive-domain-semiring* = *domain-semiring* + *involutive-dioid*

begin

notation *domain-op* (*dom*)

lemma *strong-conv-conv*: $dom\ x \leq x \cdot x^\circ \Longrightarrow x \leq x \cdot x^\circ \cdot x$
⟨*proof*⟩

end

class *involutive-dr-modal-semiring* = *dr-modal-semiring* + *involutive-dioid*

class *involutive-dr-modal-kleene-algebra* = *involutive-dr-modal-semiring* + *kleene-algebra*

4.2 Modal semirings algebras with converse

class *dr-modal-semiring-converse* = *dr-modal-semiring* + *dioid-converse*

begin

lemma *d-conv* [*simp*]: $(dom\ x)^\circ = dom\ x$
⟨*proof*⟩

lemma *cod-conv*: $(cod\ x)^\circ = cod\ x$
⟨*proof*⟩

lemma *d-conv-cod* [*simp*]: $dom\ (x^\circ) = cod\ x$
⟨*proof*⟩

lemma *cod-conv-d*: $cod\ (x^\circ) = dom\ x$
⟨*proof*⟩

lemma $dom\ y = y \Longrightarrow fd\ (x^\circ)\ y = bd\ x\ y$
⟨*proof*⟩

lemma $dom\ y = y \Longrightarrow bd\ (x^\circ)\ y = fd\ x\ y$
⟨*proof*⟩

end

4.3 Modal Kleene algebras with converse

class *dr-modal-kleene-algebra-converse* = *dr-modal-semiring-converse* + *kleene-algebra*

class *dr-modal-semiring-strong-converse* = *involutive-dr-modal-semiring* +
assumes *weak-dom-def*: $dom\ x \leq x \cdot x^\circ$
and *weak-cod-def*: $cod\ x \leq x^\circ \cdot x$

subclass (**in** *dr-modal-semiring-strong-converse*) *dr-modal-semiring-converse*
⟨*proof*⟩

class *dr-modal-kleene-algebra-strong-converse* = *dr-modal-semiring-strong-converse* + *kleene-algebra*

end

5 Modal quantales

theory *Modal-Quantale*
imports *Quantales.Quantale-Star Modal-Kleene-Algebra-Var KAD.Modal-Kleene-Algebra*

begin

5.1 Simplified modal semirings and Kleene algebras

The previous formalisation of modal Kleene algebra in the AFP adds two compatibility axioms between domain and codomain when combining an antidomain semiring with an antirange semiring. But these are unnecessary. They are derivable from the other axioms. Thus I provide a simpler axiomatisation that should eventually replace the one in the AFP.

class *modal-semiring-simp* = *antidomain-semiring* + *antirange-semiring*

lemma (**in** *modal-semiring-simp*) *dr-compat* [*simp*]: $d\ (r\ x) = r\ x$
⟨*proof*⟩

lemma (**in** *modal-semiring-simp*) *rd-compat* [*simp*]: $r\ (d\ x) = d\ x$
⟨*proof*⟩

subclass (**in** *modal-semiring-simp*) *modal-semiring*
⟨*proof*⟩

class *modal-kleene-algebra-simp* = *modal-semiring-simp* + *kleene-algebra*

subclass (**in** *modal-kleene-algebra-simp*) *modal-kleene-algebra*⟨*proof*⟩

5.2 Domain quantales

class *domain-quantale* = *unital-quantale* + *domain-op* +
assumes *dom-absorb*: $x \leq dom\ x \cdot x$
and *dom-local*: $dom\ (x \cdot dom\ y) = dom\ (x \cdot y)$
and *dom-add*: $dom\ (x \sqcup y) = dom\ x \sqcup dom\ y$
and *dom-subid*: $dom\ x \leq 1$
and *dom-zero* [*simp*]: $dom\ \bot = \bot$

The definition is that of a domain semiring. I cannot extend the quantale class with respect to domain semirings because of different operations are used for addition/sup. The following sublocale statement brings all those properties into scope.

sublocale *domain-quantale* ⊆ *dqmsr*: *domain-semiring* (⊔) (·) *1* ⊥ *dom* (≤) (<)
⟨*proof*⟩

sublocale *domain-quantale* ⊆ *dqmka*: *domain-kleene-algebra* (⊔) (·) *1* ⊥ *dom* (≤) (<) *qstar*⟨*proof*⟩

typedef (**overloaded**) *'a d-element* = {*x* :: *'a* :: *domain-quantale*. *dom x* = *x*}
⟨*proof*⟩

setup-lifting *type-definition-d-element*

instantiation *d-element* :: (*domain-quantale*) *bounded-lattice*

begin

lift-definition *less-eq-d-element* :: *'a d-element* ⇒ *'a d-element* ⇒ *bool* **is** (≤) ⟨*proof*⟩

lift-definition *less-d-element* :: *'a d-element* ⇒ *'a d-element* ⇒ *bool* **is** (<) ⟨*proof*⟩

lift-definition *bot-d-element* :: *'a d-element* **is** ⊥
⟨*proof*⟩

lift-definition *top-d-element* :: *'a d-element* **is** *1*
⟨*proof*⟩

lift-definition *inf-d-element* :: *'a d-element* ⇒ *'a d-element* ⇒ *'a d-element* **is** (·)
⟨*proof*⟩

lift-definition *sup-d-element* :: *'a d-element* ⇒ *'a d-element* ⇒ *'a d-element* **is** (⊔)
⟨*proof*⟩

instance
⟨*proof*⟩

end

instance *d-element* :: (*domain-quantale*) *distrib-lattice*
⟨*proof*⟩

context *domain-quantale*
begin

lemma *dom-top* [*simp*]: *dom* ⊤ = *1*
⟨*proof*⟩

lemma *dom-top2*: *x* · ⊤ ≤ *dom x* · ⊤
⟨*proof*⟩

lemma *weak-twisted*: $x \cdot dom\ y \leq dom\ (x \cdot y) \cdot x$
⟨*proof*⟩

lemma *dom-meet*: $dom\ x \cdot dom\ y = dom\ x \sqcap dom\ y$
⟨*proof*⟩

lemma *dom-meet-pres*: $dom\ (dom\ x \sqcap dom\ y) = dom\ x \sqcap dom\ y$
⟨*proof*⟩

lemma *dom-meet-distl*: $dom\ x \cdot (y \sqcap z) = (dom\ x \cdot y) \sqcap (dom\ x \cdot z)$
⟨*proof*⟩

lemma *dom-meet-approx*: $dom\ ((dom\ x \cdot y) \sqcap (dom\ x \cdot z)) \leq dom\ x$
⟨*proof*⟩

lemma *dom-inf-pres-aux*: $Y \neq \{\} \Longrightarrow dom\ (\bigsqcap y \in Y.\ dom\ x \cdot y) \leq dom\ x$
⟨*proof*⟩

lemma *dom-inf-pres-aux2*: $(\bigsqcap y \in Y.\ dom\ x \cdot y) \leq \bigsqcap Y$
⟨*proof*⟩

lemma *dom-inf-pres*: $Y \neq \{\} \Longrightarrow dom\ x \cdot (\bigsqcap Y) = (\bigsqcap y \in Y.\ dom\ x \cdot y)$
⟨*proof*⟩

lemma $dom\ (\bigsqcap X) \leq \bigsqcap\ (dom\ `\ X)$
⟨*proof*⟩

The domain operation need not preserve arbitrary sups, though this property holds, for instance, in quantales of binary relations. I do not aim at a stronger axiomatisation in this theory.

lemma *dom-top-pres*: $(x \leq dom\ y \cdot x) = (x \leq dom\ y \cdot \top)$
⟨*proof*⟩

lemma *dom-lla-var*: $(dom\ x \leq dom\ y) = (x \leq dom\ y \cdot \top)$
⟨*proof*⟩

lemma $dom\ (1 \sqcap x) = 1 \sqcap x \Longrightarrow x \leq 1 \Longrightarrow dom\ x = x$
⟨*proof*⟩

lemma *dom-meet-sub*: $dom\ (x \sqcap y) \leq dom\ x \sqcap dom\ y$
⟨*proof*⟩

lemma *dom-dist1*: $dom\ x \sqcup (dom\ y \sqcap dom\ z) = (dom\ x \sqcup dom\ y) \sqcap (dom\ x \sqcup dom\ z)$
⟨*proof*⟩

lemma *dom-dist2*: $dom\ x \sqcap (dom\ y \sqcup dom\ z) = (dom\ x \sqcap dom\ y) \sqcup (dom\ x \sqcap dom\ z)$

⟨*proof*⟩

abbreviation $fd' \equiv dqmsr.fd$

definition $bb\ x\ y = \bigsqcup\{dom\ z\ |z.\ fd'\ x\ z \leq dom\ y\}$

lemma *fd′-bb-galois-aux*: $fd'\ x\ (dom\ p) \leq dom\ q \Longrightarrow dom\ p \leq bb\ x\ (dom\ q)$
⟨*proof*⟩

lemma *dom-iso-var*: $(\bigsqcup x \in X.\ dom\ x) \leq dom\ (\bigsqcup x \in X.\ dom\ x)$
⟨*proof*⟩

lemma *dom-iso-var2*: $(\bigsqcup x \in X.\ dom\ x) \leq dom\ (\bigsqcup x \in X.\ x)$
⟨*proof*⟩

end

5.3 Codomain quantales

class *codomain-quantale* = *unital-quantale* + *range-op* +
assumes *cod-absorb*: $x \leq x \cdot cod\ x$
and *cod-local*: $cod\ (cod\ x \cdot y) = cod\ (x \cdot y)$
and *cod-add*: $cod\ (x \sqcup y) = cod\ x \sqcup cod\ y$
and *cod-subid*: $cod\ x \leq 1$
and *cod-zero*: $cod\ \bot = \bot$

sublocale *codomain-quantale* ⊆ *coddual*: *domain-quantale range-op* - $\lambda x\ y.\ y \cdot x$ -
\- - - - - - -
⟨*proof*⟩

abbreviation (**in** *codomain-quantale*) $bd' \equiv coddual.fd'$

definition (**in** *codomain-quantale*) $fb\ x\ y = \bigsqcup\{cod\ z\ |z.\ bd'\ x\ z \leq cod\ y\}$

lemma (**in** *codomain-quantale*) *bd′-fb-galois-aux*: $bd'\ x\ (cod\ p) \leq cod\ q \Longrightarrow cod\ p \leq fb\ x\ (cod\ q)$
⟨*proof*⟩

5.4 Modal quantales

class *dc-modal-quantale* = *domain-quantale* + *codomain-quantale* +
assumes *dc-compat* [*simp*]: $dom\ (cod\ x) = cod\ x$
and *cd-compat* [*simp*]: $cod\ (dom\ x) = dom\ x$

sublocale *dc-modal-quantale* ⊆ *mqs*: *dr-modal-kleene-algebra* $(\sqcup)$ $(\cdot)$ 1 $\bot$ $(\leq)$ $(<)$ *qstar dom cod*
⟨*proof*⟩

sublocale *dc-modal-quantale* ⊆ *mqdual*: *dc-modal-quantale* - $\lambda x\ y.\ y \cdot x$ - - - - - -
\- - *dom cod*

⟨*proof*⟩

lemma (**in** *dc-modal-quantale*) $x \cdot \top = dom\ x \cdot \top$

⟨*proof*⟩

lemma (**in** *dc-modal-quantale*) $\top \cdot x = \top \cdot cod\ x$

⟨*proof*⟩

5.5 Antidomain and anticodomain quantales

notation *antidomain-op* (*adom*)

class *antidomain-quantale* = *unital-quantale* + *antidomain-op* +
assumes *as1* [*simp*]: $adom\ x \cdot x = \bot$
and *as2* [*simp*]: $adom\ (x \cdot y) \leq adom\ (x \cdot adom\ (adom\ y))$
and *as3* [*simp*]: $adom\ (adom\ x) \sqcup adom\ x = 1$

definition (**in** *antidomain-quantale*) $ddom = adom \circ adom$

sublocale *antidomain-quantale* ⊆ *adqmsr*: *antidomain-semiring adom* (⊔) (·) *1* ⊥ (≤) (<)
⟨*proof*⟩

sublocale *antidomain-quantale* ⊆ *adqmka*: *antidomain-kleene-algebra adom* (⊔) (·) *1* ⊥ (≤) (<) *qstar*⟨*proof*⟩

sublocale *antidomain-quantale* ⊆ *addq*: *domain-quantale ddom*
⟨*proof*⟩

notation *antirange-op* (*acod*)

class *anticodomain-quantale* = *unital-quantale* + *antirange-op* +
assumes *ars1* [*simp*]: $x \cdot acod\ x = \bot$
and *ars2* [*simp*]: $acod\ (x \cdot y) \leq acod\ (acod\ (acod\ x) \cdot y)$
and *ars3* [*simp*]: $acod\ (acod\ x) \sqcup acod\ x = 1$

sublocale *anticodomain-quantale* ⊆ *acoddual*: *antidomain-quantale acod* - $\lambda x\ y.\ y \cdot x$ - - - - - - - -
⟨*proof*⟩

definition (**in** *anticodomain-quantale*) $ccod = acod \circ acod$

sublocale *anticodomain-quantale* ⊆ *acdqmsr*: *antirange-semiring* (⊔) (·) *1* ⊥ *acod* (≤) (<)⟨*proof*⟩

sublocale *anticodomain-quantale* ⊆ *acdqmka*: *antirange-kleene-algebra* (⊔) (·) *1* ⊥ (≤) (<) *qstar acod*⟨*proof*⟩

sublocale *anticodomain-quantale* ⊆ *acddq*: *codomain-quantale* - - - - - - - - - - λ *x*. *acod* (*acod* *x*)
⟨*proof*⟩

class *modal-quantale* = *antidomain-quantale* + *anticodomain-quantale*

sublocale *modal-quantale* ⊆ *mmqs*: *modal-kleene-algebra-simp* (⊔) (·) *1* ⊥ (≤) (<) *qstar adom acod*⟨*proof*⟩

sublocale *modal-quantale* ⊆ *mmqdual*: *modal-quantale* - $\lambda x\ y.\ y \cdot x$ - - - - - - - - *adom acod*
⟨*proof*⟩

end

6 Quantales with converse

theory *Quantale-Converse*
imports *Modal-Quantale Modal-Kleene-Algebra-Converse*

begin

6.1 Properties of unital quantales

These properties should eventually added to the quantales AFP entry.

lemma (**in** *quantale*) *bres-bot-top* [*simp*]: $\bot \to \top = \top$
⟨*proof*⟩

lemma (**in** *quantale*) *fres-top-bot* [*simp*]: $\top \leftarrow \bot = \top$
⟨*proof*⟩

lemma (**in** *unital-quantale*) *bres-top-top2* [*simp*]: $(x \to y \cdot \top) \cdot \top = x \to y \cdot \top$
⟨*proof*⟩

lemma (**in** *unital-quantale*) *fres-top-top2* [*simp*]: $\top \cdot (\top \cdot y \leftarrow x) = \top \cdot y \leftarrow x$
⟨*proof*⟩

lemma (**in** *unital-quantale*) *bres-top-bot* [*simp*]: $\top \to \bot = \bot$
⟨*proof*⟩

lemma (**in** *unital-quantale*) *fres-bot-top* [*simp*]: $\bot \leftarrow \top = \bot$
⟨*proof*⟩

lemma (**in** *unital-quantale*) *top-bot-iff*: $(x \cdot \top = \bot) = (x = \bot)$
⟨*proof*⟩

6.2 Involutive quantales

The following axioms for involutive quantales are standard.

class *involutive-quantale* = *unital-quantale* + *invol-op* +
assumes *inv-invol* [*simp*]: $(x^\circ)^\circ = x$
and *inv-contrav*: $(x \cdot y)^\circ = y^\circ \cdot x^\circ$
and *inv-sup* [*simp*]: $(\bigsqcup X)^\circ = (\bigsqcup x \in X.\ x^\circ)$

context *involutive-quantale*
begin

lemma *inv-binsup* [*simp*]: $(x \sqcup y)^\circ = x^\circ \sqcup y^\circ$
⟨*proof*⟩

lemma *inv-iso*: $x \leq y \Longrightarrow x^\circ \leq y^\circ$
⟨*proof*⟩

lemma *inv-galois*: $(x^\circ \leq y) = (x \leq y^\circ)$
⟨*proof*⟩

lemma *bres-fres-conv*: $(y^\circ \leftarrow x^\circ)^\circ = x \rightarrow y$
⟨*proof*⟩

lemma *fres-bres-conv*: $(y^\circ \rightarrow x^\circ)^\circ = x \leftarrow y$
⟨*proof*⟩

sublocale *invqka*: *involutive-kleene-algebra* $(\sqcup)$ $(\cdot)$ *1* $\bot$ $(\leq)$ $(<)$ *qstar invol*
⟨*proof*⟩

lemma *inv-binf* [*simp*]: $(x \sqcap y)^\circ = x^\circ \sqcap y^\circ$
⟨*proof*⟩

lemma *inv-inf* [*simp*]: $(\bigsqcap X)^\circ = (\bigsqcap x \in X.\ x^\circ)$
⟨*proof*⟩

lemma *inv-top* [*simp*]: $\top^\circ = \top$
⟨*proof*⟩

lemma *inv-qstar-aux* [*simp*]: $(x \frown i)^\circ = (x^\circ) \frown i$
⟨*proof*⟩

lemma *inv-conjugate*: $(x^\circ \sqcap y = \bot) = (x \sqcap y^\circ = \bot)$
⟨*proof*⟩

We define domain and codomain as in relation algebra and compare with the domain and codomain axioms above.

definition *do* :: $'a \Rightarrow 'a$ **where**
do $x = 1 \sqcap (x \cdot x^\circ)$

definition *cd* :: $'a \Rightarrow 'a$ **where**
  $cd\ x = 1 \sqcap (x^\circ \cdot x)$

lemma *do-inv*: $do\ (x^\circ) = cd\ x$
⟨*proof*⟩

lemma *cd-inv*: $cd\ (x^\circ) = do\ x$
  ⟨*proof*⟩

lemma *do-le-top*: $do\ x \leq 1 \sqcap (x \cdot \top)$
  ⟨*proof*⟩

lemma *do-subid*: $do\ x \leq 1$
  ⟨*proof*⟩

lemma *cd-subid*: $cd\ x \leq 1$
  ⟨*proof*⟩

lemma *do-bot* [*simp*]: $do\ \bot = \bot$
  ⟨*proof*⟩

lemma *cd-bot* [*simp*]: $cd\ \bot = \bot$
  ⟨*proof*⟩

lemma *do-iso*: $x \leq y \Longrightarrow do\ x \leq do\ y$
  ⟨*proof*⟩

lemma *cd-iso*: $x \leq y \Longrightarrow cd\ x \leq cd\ y$
  ⟨*proof*⟩

lemma *do-subdist*: $do\ x \sqcup do\ y \leq do\ (x \sqcup y)$
⟨*proof*⟩

lemma *cd-subdist*: $cd\ x \sqcup cd\ y \leq cd\ (x \sqcup y)$
  ⟨*proof*⟩

lemma *inv-do* [*simp*]: $(do\ x)^\circ = do\ x$
  ⟨*proof*⟩

lemma *inv-cd* [*simp*]: $(cd\ x)^\circ = cd\ x$
  ⟨*proof*⟩

lemma *dedekind-modular*:
  assumes $(x \cdot y) \sqcap z \leq (x \sqcap (z \cdot y^\circ)) \cdot (y \sqcap (x^\circ \cdot z))$
  shows $(x \cdot y) \sqcap z \leq (x \sqcap (z \cdot y^\circ)) \cdot y$
  ⟨*proof*⟩

lemma *modular-eq1*:
  assumes $\forall x\ y\ z\ w.\ (y \sqcap (z \cdot x^\circ) \leq w \longrightarrow (y \cdot x) \sqcap z \leq w \cdot x)$

shows $\forall x\ y\ z.\ (x \cdot y) \sqcap z \le (x \sqcap (z \cdot y^\circ)) \cdot y$
⟨*proof*⟩

lemma $do\ x \cdot do\ y = do\ x \sqcap do\ y$
⟨*proof*⟩

lemma $p \le 1 \Longrightarrow q \le 1 \Longrightarrow p \cdot q = p \sqcap q$
⟨*proof*⟩

end

sublocale *ab-unital-quantale* ⊆ *ciq*: *involutive-quantale id* - - - - - - - - - -
⟨*proof*⟩

class *distributive-involutive-quantale* = *involutive-quantale* + *distrib-unital-quantale*

class *boolean-involutive-quantale* = *involutive-quantale* + *bool-unital-quantale*

begin

lemma *res-peirce*:
assumes $\forall x\ y.\ x^\circ \cdot -(x \cdot y) \le -y$
shows $((x \cdot y) \sqcap z^\circ = \bot) = ((y \cdot z) \sqcap x^\circ = \bot)$
⟨*proof*⟩

lemma *res-schroeder1*:
assumes $\forall x\ y.\ x^\circ \cdot -(x \cdot y) \le -y$
shows $((x \cdot y) \sqcap z = \bot) = (y \sqcap (x^\circ \cdot z) = \bot)$
⟨*proof*⟩

lemma *res-schroeder2*:
assumes $\forall x\ y.\ x^\circ \cdot -(x \cdot y) \le -y$
shows $((x \cdot y) \sqcap z = \bot) = (x \sqcap (z \cdot y^\circ) = \bot)$
⟨*proof*⟩

lemma *res-mod*:
assumes $\forall x\ y.\ x^\circ \cdot -(x \cdot y) \le -y$
shows $(x \cdot y) \sqcap z \le (x \sqcap (z \cdot y^\circ)) \cdot y$
⟨*proof*⟩

end

The strong Gelfand property (name by Palmigiano and Re) is important for dioids and Kleene algebras. The modular law is a convenient axiom for relational quantales, in a setting where the underlying lattice is not boolean.

class *quantale-converse* = *involutive-quantale* +
assumes *strong-gelfand*: $x \le x \cdot x^\circ \cdot x$

begin

lemma *do-gelfand* [*simp*]: $do\ x \cdot do\ x \cdot do\ x = do\ x$
⟨*proof*⟩

lemma *cd-gelfand* [*simp*]: $cd\ x \cdot cd\ x \cdot cd\ x = cd\ x$
⟨*proof*⟩

lemma *do-idem* [*simp*]: $do\ x \cdot do\ x = do\ x$
⟨*proof*⟩

lemma *cd-idem* [*simp*]: $cd\ x \cdot cd\ x = cd\ x$
⟨*proof*⟩

lemma *dodo* [*simp*]: $do\ (do\ x) = do\ x$
⟨*proof*⟩

lemma *cdcd* [*simp*]: $cd\ (cd\ x) = cd\ x$
⟨*proof*⟩

lemma *docd-compat* [*simp*]: $do\ (cd\ x) = cd\ x$
⟨*proof*⟩

lemma *cddo-compat* [*simp*]: $cd\ (do\ x) = do\ x$
⟨*proof*⟩

end

sublocale *quantale-converse* ⊆ *convqka*: *kleene-algebra-converse* $(\sqcup)$ $(\cdot)$ *1* $\bot$ $(\leq)$ $(<)$ *invol qstar*
⟨*proof*⟩

6.3 Dedekind quantales

class *dedekind-quantale* = *involutive-quantale* +
assumes *modular-law*: $(x \cdot y) \sqcap z \leq (x \sqcap (z \cdot y^\circ)) \cdot y$

begin

sublocale *convdqka*: *kleene-algebra-converse* $(\sqcup)$ $(\cdot)$ *1* $\bot$ $(\leq)$ $(<)$ *invol qstar*
⟨*proof*⟩

subclass *quantale-converse*
⟨*proof*⟩

lemma *modular-2* [*simp*]: $((x \sqcap (z \cdot y^\circ)) \cdot y) \sqcap z = (x \cdot y) \sqcap z$
⟨*proof*⟩

lemma *modular-1* [*simp*]: $(x \cdot (y \sqcap (x^\circ \cdot z))) \sqcap z = (x \cdot y) \sqcap z$
⟨*proof*⟩

lemma *modular3*: $(x \cdot y) \sqcap z \leq x \cdot (y \sqcap (x^\circ \cdot z))$
⟨*proof*⟩

The name Dedekind quantale owes to the following formula, which is equivalent to the modular law. Dedekind quantales are called modular quantales in Rosenthal's book on quantaloids (to be precise: he discusses modular quantaloids, but the notion of modular quantale is then obvious).

lemma *dedekind*: $(x \cdot y) \sqcap z \leq (x \sqcap (z \cdot y^\circ)) \cdot (y \sqcap (x^\circ \cdot z))$
⟨*proof*⟩

lemma *peirce*: $((x \cdot y) \sqcap z^\circ = \bot) = ((y \cdot z) \sqcap x^\circ = \bot)$
⟨*proof*⟩

lemma *schroeder-1*: $((x \cdot y) \sqcap z = \bot) = (y \sqcap (x^\circ \cdot z) = \bot)$
⟨*proof*⟩

lemma *schroeder-2*: $((x \cdot y) \sqcap z = \bot) = (x \sqcap (z \cdot y^\circ) = \bot)$
⟨*proof*⟩

lemma *modular-eq2*: $y \sqcap (z \cdot x^\circ) \leq w \Longrightarrow (y \cdot x) \sqcap z \leq w \cdot x$
⟨*proof*⟩

lemma *lla-top-aux*: $p \leq 1 \Longrightarrow ((x \leq p \cdot x) = (x \leq p \cdot \top))$
⟨*proof*⟩

Next we turn to properties of domain and codomain in Dedekind quantales.

lemma *lra-top-aux*: $p \leq 1 \Longrightarrow ((x \leq x \cdot p\ ) = (x \leq \top \cdot p))$
⟨*proof*⟩

lemma *lla*: $p \leq 1 \Longrightarrow ((do\ x \leq p) = (x \leq p \cdot \top))$
⟨*proof*⟩

lemma *lla-Inf*: $do\ x = \bigsqcap\{p.\ x \leq p \cdot \top \wedge p \leq 1\}$
⟨*proof*⟩

lemma *lra*: $p \leq 1 \Longrightarrow ((cd\ x \leq p) = (x \leq \top \cdot p))$
⟨*proof*⟩

lemma *lra-Inf*: $cd\ x = \bigsqcap\{p.\ x \leq \top \cdot p \wedge p \leq 1\}$
⟨*proof*⟩

lemma *lla-var*: $p \leq 1 \Longrightarrow ((do\ x \leq p) = (x \leq p \cdot x))$
⟨*proof*⟩

lemma *lla-Inf-var*: $do\ x = \bigsqcap\{p.\ x \leq p \cdot x \wedge p \leq 1\}$
⟨*proof*⟩

lemma *lra-var*: $p \leq 1 \Longrightarrow ((cd\ x \leq p) = (x \leq x \cdot p))$

⟨*proof*⟩

lemma *lra-Inf-var*: $cd\ x = \bigsqcap\{p.\ x \leq x \cdot p \wedge p \leq 1\}$
⟨*proof*⟩

lemma *do-top*: $do\ x = 1 \sqcap (x \cdot \top)$
⟨*proof*⟩

lemma *cd-top*: $cd\ x = 1 \sqcap (\top \cdot x)$
⟨*proof*⟩

We start deriving the axioms of modal semirings and modal quantales.

lemma *do-absorp*: $x \leq do\ x \cdot x$
⟨*proof*⟩

lemma *cd-absorp*: $x \leq x \cdot cd\ x$
⟨*proof*⟩

lemma *do-absorp-eq* [*simp*]: $do\ x \cdot x = x$
⟨*proof*⟩

lemma *cd-absorp-eq* [*simp*]: $x \cdot cd\ x = x$
⟨*proof*⟩

lemma *do-top2*: $x \cdot \top = do\ x \cdot \top$
⟨*proof*⟩

lemma *cd-top2*: $\top \cdot x = \top \cdot cd\ x$
⟨*proof*⟩

lemma *do-local* [*simp*]: $do\ (x \cdot do\ y) = do\ (x \cdot y)$
⟨*proof*⟩

lemma *cd-local* [*simp*]: $cd\ (cd\ x \cdot y) = cd\ (x \cdot y)$
⟨*proof*⟩

lemma *do-fix-subid*: $(do\ x = x) = (x \leq 1)$
⟨*proof*⟩

lemma *cd-fix-subid*: $(cd\ x = x) = (x \leq 1)$
⟨*proof*⟩

lemma *do-inf2*: $do\ (do\ x \sqcap do\ y) = do\ x \sqcap do\ y$
⟨*proof*⟩

lemma *do-inf-comp*: $do\ x \cdot do\ y = do\ x \sqcap do\ y$
⟨*proof*⟩

lemma *cd-inf-comp*: $cd\ x \cdot cd\ y = cd\ x \sqcap cd\ y$

⟨*proof*⟩

lemma *subid-mult-meet*: $p \leq 1 \Longrightarrow q \leq 1 \Longrightarrow p \cdot q = p \sqcap q$
⟨*proof*⟩

lemma *dodo-sup*: $do\ (do\ x \sqcup do\ y) = do\ x \sqcup do\ y$
⟨*proof*⟩

lemma *do-sup* [*simp*]: $do\ (x \sqcup y) = do\ x \sqcup do\ y$
⟨*proof*⟩

lemma *cdcd-sup*: $cd\ (cd\ x \sqcup cd\ y) = cd\ x \sqcup cd\ y$
⟨*proof*⟩

lemma *cd-sup* [*simp*]: $cd\ (x \sqcup y) = cd\ x \sqcup cd\ y$
⟨*proof*⟩

Next we show that Dedekind quantales are modal quantales, hence also modal semirings.

sublocale *dmq*: *dc-modal-quantale 1* $(\cdot)$ *Inf Sup* $(\sqcap)$ $(\leq)$ $(<)$ $(\sqcup)$ $\bot$ $\top$ *cd do*
⟨*proof*⟩

lemma *do-top3* [*simp*]: $do\ (x \cdot \top) = do\ x$
⟨*proof*⟩

lemma *cd-top3* [*simp*]: $cd\ (\top \cdot x) = cd\ x$
⟨*proof*⟩

lemma *dodo-Sup-pres*: $do\ (\bigsqcup x \in X.\ do\ x) = (\bigsqcup x \in X.\ do\ x)$
⟨*proof*⟩

The domain elements form a complete Heyting algebra.

lemma *do-complete-heyting*: $do\ x \sqcap (\bigsqcup y \in Y.\ do\ y) = (\bigsqcup y \in Y.\ do\ x \sqcap do\ y)$
⟨*proof*⟩

lemma *cdcd-Sup-pres*: $cd\ (\bigsqcup x \in X.\ cd\ x) = (\bigsqcup x \in X.\ cd\ x)$
⟨*proof*⟩

lemma *cd-complete-heyting*: $cd\ x \sqcap (\bigsqcup y \in Y.\ cd\ y) = (\bigsqcup y \in Y.\ cd\ x \sqcap cd\ y)$
⟨*proof*⟩

lemma *subid-complete-heyting*:
assumes $p \leq 1$
and $\forall q \in Q.\ q \leq 1$
shows $p \sqcap (\bigsqcup Q) = (\bigsqcup q \in Q.\ p \sqcap q)$
⟨*proof*⟩

Next we show that domain and codomain preserve arbitrary Sups.

lemma *do-Sup-pres-aux*: $(\bigsqcup x \in X.\ do\ x \cdot \top) = (\bigsqcup x \in do\ `\ X.\ x \cdot \top)$

⟨*proof*⟩

lemma *do-Sup-pres*: $do\ (\bigsqcup x \in X.\ x) = (\bigsqcup x \in X.\ do\ x)$
⟨*proof*⟩

lemma *cd-Sup-pres*: $cd\ (\bigsqcup x \in X.\ x) = (\bigsqcup x \in X.\ cd\ x)$
⟨*proof*⟩

lemma *do-inf*: $do\ (x \sqcap y) = 1 \sqcap (y \cdot x^\circ)$
⟨*proof*⟩

lemma *cd-inf*: $cd\ (x \sqcap y) = 1 \sqcap (y^\circ \cdot x)$
⟨*proof*⟩

lemma *do-bres-prop*: $p \le 1 \Longrightarrow do\ (x \to p \cdot \top) = 1 \sqcap (x \to p \cdot \top)$
⟨*proof*⟩

lemma *cd-fres-prop*: $p \le 1 \Longrightarrow cd\ (\top \cdot p \leftarrow x) = 1 \sqcap (\top \cdot p \leftarrow x)$
⟨*proof*⟩

lemma *do-meet-prop*: $(do\ p \cdot x) \sqcap (x \cdot do\ q) = do\ p \cdot x \cdot do\ q$
⟨*proof*⟩

lemma *subid-meet-prop*: $p \le 1 \Longrightarrow q \le 1 \Longrightarrow (p \cdot x) \sqcap (x \cdot q) = p \cdot x \cdot q$
⟨*proof*⟩

Next we consider box and diamond operators like in modal semirings and modal quantales. These are inherited from domain quantales. Diamonds are defined with respect to domain and codomain. The box operators are defined as Sups and hence right adjoints of diamonds.

abbreviation *do-dia* ≡ *dmq.fd′*

abbreviation *cd-dia* ≡ *dmq.bd′*

abbreviation *do-box* ≡ *dmq.bb*

abbreviation *cd-box* ≡ *dmq.fb*

In the sense of modal logic, the domain-based diamond is a backward operator, the codomain-based one a forward operator. These are related by opposition/converse.

lemma *do-dia-cd-dia-conv*: $p \le 1 \Longrightarrow \textit{do-dia}\ (x^\circ)\ p = \textit{cd-dia}\ x\ p$
⟨*proof*⟩

lemma *cd-dia-do-dia-conv*: $p \le 1 \Longrightarrow \textit{cd-dia}\ (x^\circ)\ p = \textit{do-dia}\ x\ p$
⟨*proof*⟩

Diamonds preserve sups in both arguments.

lemma *do-dia-Sup*: *do-dia* $(\bigsqcup X)$ *p* = $(\bigsqcup x \in X.$ *do-dia x p*)
⟨*proof*⟩

lemma *do-dia-Sup2*: *do-dia x* $(\bigsqcup P) = (\bigsqcup p \in P.$ *do-dia x p*)
⟨*proof*⟩

lemma *cd-dia-Sup*: *cd-dia* $(\bigsqcup X)$ *p* = $(\bigsqcup x \in X.$ *cd-dia x p*)
⟨*proof*⟩

lemma *cd-dia-Sup2*: *cd-dia x* $(\bigsqcup P) = (\bigsqcup p \in P.$ *cd-dia x p*)
⟨*proof*⟩

The domain-based box is a forward operator, the codomain-based on a backward one. These interact again with respect to converse.

lemma *do-box-var*: $p \leq 1 \Longrightarrow$ *do-box x p* = $\bigsqcup\{q.$ *do-dia x q* $\leq p \wedge q \leq 1\}$
⟨*proof*⟩

lemma *cd-box-var*: $p \leq 1 \Longrightarrow$ *cd-box x p* = $\bigsqcup\{q.$ *cd-dia x q* $\leq p \wedge q \leq 1\}$
⟨*proof*⟩

lemma *do-box-cd-box-conv*: $p \leq 1 \Longrightarrow$ *do-box* (x°) *p* = *cd-box x p*
⟨*proof*⟩

lemma *cd-box-do-box-conv*: $p \leq 1 \Longrightarrow$ *cd-box* (x°) *p* = *do-box x p*
⟨*proof*⟩

lemma *do-box-subid*: *do-box x p* ≤ 1
⟨*proof*⟩

lemma *cd-box-subid*: $p \leq 1 \Longrightarrow$ *cd-box x p* ≤ 1
⟨*proof*⟩

Next we prove that boxes and diamonds are adjoints, and then demodalisation laws known from modal semirings.

lemma *do-dia-do-box-galois*:
assumes $p \leq 1$
and $q \leq 1$
shows (*do-dia x p* $\leq q$) = ($p \leq$ *do-box x q*)
⟨*proof*⟩

lemma *cd-dia-cd-box-galois*:
assumes $p \leq 1$
and $q \leq 1$
shows (*cd-dia x p* $\leq q$) = ($p \leq$ *cd-box x q*)
⟨*proof*⟩

lemma *do-dia-demod-subid*:
assumes $p \leq 1$
and $q \leq 1$

shows $(do\text{-}dia\ x\ p \leq q) = (x \cdot p \leq q \cdot x)$
⟨*proof*⟩

The demodalisation laws have variants based on residuals.

lemma *do-dia-demod-subid-fres*:
assumes $p \leq 1$
and $q \leq 1$
shows $(do\text{-}dia\ x\ p \leq q) = (p \leq x \rightarrow q \cdot x)$
⟨*proof*⟩

lemma *do-dia-demod-subid-var*:
assumes $p \leq 1$
and $q \leq 1$
shows $(do\text{-}dia\ x\ p \leq q) = (x \cdot p \leq q \cdot \top)$
⟨*proof*⟩

lemma *do-dia-demod-subid-var-fres*:
assumes $p \leq 1$
and $q \leq 1$
shows $(do\text{-}dia\ x\ p \leq q) = (p \leq x \rightarrow q \cdot \top)$
⟨*proof*⟩

lemma *cd-dia-demod-subid*:
assumes $p \leq 1$
and $q \leq 1$
shows $(cd\text{-}dia\ x\ p \leq q) = (p \cdot x \leq x \cdot q)$
⟨*proof*⟩

lemma *cd-dia-demod-subid-fres*:
assumes $p \leq 1$
and $q \leq 1$
shows $(cd\text{-}dia\ x\ p \leq q) = (p \leq x \cdot q \leftarrow x)$
⟨*proof*⟩

lemma *cd-dia-demod-subid-var*:
assumes $p \leq 1$
and $q \leq 1$
shows $(cd\text{-}dia\ x\ p \leq q) = (p \cdot x \leq \top \cdot q)$
⟨*proof*⟩

lemma *cd-dia-demod-subid-var-fres*:
assumes $p \leq 1$
and $q \leq 1$
shows $(cd\text{-}dia\ x\ p \leq q) = (p \leq \top \cdot q \leftarrow x)$
⟨*proof*⟩

lemma *do-box-iso*:
assumes $p \leq 1$
and $q \leq 1$

and $p \leq q$
shows *do-box* x $p \leq$ *do-box* x q
⟨*proof*⟩

lemma *cd-box-iso*:
assumes $p \leq 1$
and $q \leq 1$
and $p \leq q$
shows *cd-box* x $p \leq$ *cd-box* x q
⟨*proof*⟩

lemma *do-box-demod-subid*:
assumes $p \leq 1$
and $q \leq 1$
shows ($p \leq$ *do-box* x q) $=$ ($x \cdot p \leq q \cdot x$)
⟨*proof*⟩

lemma *do-box-demod-subid-bres*:
assumes $p \leq 1$
and $q \leq 1$
shows ($p \leq$ *do-box* x q) $=$ ($p \leq x \rightarrow q \cdot x$)
⟨*proof*⟩

lemma *do-box-demod-subid-var*:
assumes $p \leq 1$
and $q \leq 1$
shows ($p \leq$ *do-box* x q) $=$ ($x \cdot p \leq q \cdot \top$)
⟨*proof*⟩

lemma *do-box-demod-subid-var-bres*:
assumes $p \leq 1$
and $q \leq 1$
shows ($p \leq$ *do-box* x q) $=$ ($p \leq x \rightarrow q \cdot \top$)
⟨*proof*⟩

lemma *do-box-demod-subid-var-bres-do*:
assumes $p \leq 1$
and $q \leq 1$
shows ($p \leq$ *do-box* x q) $=$ ($p \leq$ *do* ($x \rightarrow q \cdot \top$))
⟨*proof*⟩

lemma *cd-box-demod-subid*:
assumes $p \leq 1$
and $q \leq 1$
shows ($p \leq$ *cd-box* x q) $=$ ($p \cdot x \leq x \cdot q$)
⟨*proof*⟩

lemma *cd-box-demod-subid-fres*:
assumes $p \leq 1$

and $q \leq 1$
shows $(p \leq \textit{cd-box } x\ q) = (p \leq x \cdot q \leftarrow x)$
⟨*proof*⟩

lemma *cd-box-demod-subid-var*:
assumes $p \leq 1$
and $q \leq 1$
shows $(p \leq \textit{cd-box } x\ q) = (p \cdot x \leq \top \cdot q)$
⟨*proof*⟩

lemma *cd-box-demod-subid-var-fres*:
assumes $p \leq 1$
and $q \leq 1$
shows $(p \leq \textit{cd-box } x\ q) = (p \leq \top \cdot q \leftarrow x)$
⟨*proof*⟩

We substitute demodalisation inequalities for diamonds in the definitions of boxes.

lemma *do-box-var2*: $p \leq 1 \Longrightarrow \textit{do-box } x\ p = \bigsqcup\{q.\ x \cdot q \leq p \cdot x \wedge q \leq 1\}$
⟨*proof*⟩

lemma *do-box-bres1*: $p \leq 1 \Longrightarrow \textit{do-box } x\ p = \bigsqcup\{q.\ q \leq x \rightarrow p \cdot x \wedge q \leq 1\}$
⟨*proof*⟩

lemma *do-box-bres2*: $p \leq 1 \Longrightarrow \textit{do-box } x\ p = \bigsqcup\{q.\ q \leq x \rightarrow p \cdot \top \wedge q \leq 1\}$
⟨*proof*⟩

lemma *do-box-var3*: $p \leq 1 \Longrightarrow \textit{do-box } x\ p = \bigsqcup\{q.\ x \cdot q \leq p \cdot \top \wedge q \leq 1\}$
⟨*proof*⟩

lemma *cd-box-var2*: $p \leq 1 \Longrightarrow \textit{cd-box } x\ p = \bigsqcup\{q.\ q \cdot x \leq x \cdot p \wedge q \leq 1\}$
⟨*proof*⟩

lemma *cd-box-var3*: $p \leq 1 \Longrightarrow \textit{cd-box } x\ p = \bigsqcup\{q.\ q \cdot x \leq \top \cdot p \wedge q \leq 1\}$
⟨*proof*⟩

Using these results we get a simple characterisation of boxes via domain and codomain. Similar laws can be found implicitly in Doornbos, Backhouse and van der Woude's paper on a calculational approach to mathematical induction, which speaks about wlp operators instead modal operators.

lemma *bres-do-box*: $p \leq 1 \Longrightarrow \textit{do-box } x\ p = \textit{do } (x \rightarrow p \cdot \top)$
⟨*proof*⟩

lemma *bres-do-box-var*: $p \leq 1 \Longrightarrow \textit{do-box } x\ p = 1 \sqcap (x \rightarrow p \cdot \top)$
⟨*proof*⟩

lemma *bres-do-box-top*: $p \leq 1 \Longrightarrow (\textit{do-box } x\ p) \cdot \top = x \rightarrow p \cdot \top$
⟨*proof*⟩

lemma *fres-cd-box*: $p \leq 1 \Longrightarrow cd\text{-}box\ x\ p = cd\ (\top \cdot p \leftarrow x)$
⟨*proof*⟩

lemma *fres-cd-box-var*: $p \leq 1 \Longrightarrow cd\text{-}box\ x\ p = 1 \sqcap (\top \cdot p \leftarrow x)$
⟨*proof*⟩

lemma *fres-cd-box-top*: $p \leq 1 \Longrightarrow \top \cdot cd\text{-}box\ x\ p = \top \cdot p \leftarrow x$
⟨*proof*⟩

Next we show that the box operators act on the complete Heyting algebra of subidentities.

lemma *cd-box-act*:
assumes $p \leq 1$
shows $cd\text{-}box\ (x \cdot y)\ p = cd\text{-}box\ x\ (cd\text{-}box\ y\ p)$
⟨*proof*⟩

lemma *do-box-act*:
assumes $p \leq 1$
shows $do\text{-}box\ (x \cdot y)\ p = do\text{-}box\ y\ (do\text{-}box\ x\ p)$
⟨*proof*⟩

Next we show that the box operators are Sup reversing in the first and Inf preserving in the second argument.

lemma *do-box-sup-inf*: $p \leq 1 \Longrightarrow do\text{-}box\ (x \sqcup y)\ p = do\text{-}box\ x\ p \cdot do\text{-}box\ y\ p$
⟨*proof*⟩

lemma *do-box-sup-inf-var*: $p \leq 1 \Longrightarrow do\text{-}box\ (x \sqcup y)\ p = do\text{-}box\ x\ p \sqcap do\text{-}box\ y\ p$
⟨*proof*⟩

lemma *do-box-Sup-Inf*:
assumes $X \neq \{\}$
and $p \leq 1$
shows $do\text{-}box\ (\bigsqcup X)\ p = (\bigsqcap x \in X.\ do\text{-}box\ x\ p)$
⟨*proof*⟩

lemma *do-box-Sup-Inf2*:
assumes $P \neq \{\}$
and $\forall p \in P.\ p \leq 1$
shows $do\text{-}box\ x\ (\bigsqcap P) = (\bigsqcap p \in P.\ do\text{-}box\ x\ p)$
⟨*proof*⟩

lemma *cd-box-sup-inf*: $p \leq 1 \Longrightarrow cd\text{-}box\ (x \sqcup y)\ p = cd\text{-}box\ x\ p \cdot cd\text{-}box\ y\ p$
⟨*proof*⟩

lemma *cd-box-sup-inf-var*: $p \leq 1 \Longrightarrow cd\text{-}box\ (x \sqcup y)\ p = cd\text{-}box\ x\ p \sqcap cd\text{-}box\ y\ p$
⟨*proof*⟩

lemma *cd-box-Sup-Inf*:
assumes $X \neq \{\}$

and $p \leq 1$
shows *cd-box* $(\bigsqcup X)$ $p = (\bigsqcap x \in X.$ *cd-box* x $p)$
⟨*proof*⟩

lemma *cd-box-Sup-Inf2*:
assumes $P \neq \{\}$
and $\forall p \in P.\ p \leq 1$
shows *cd-box* x $(\bigsqcap P) = (\bigsqcap p \in P.$ *cd-box* x $p)$
⟨*proof*⟩

Next we define an antidomain operation in the style of modal semirings. A natural condition is that the antidomain of an element is the greatest test that cannot be left-composed with that elements, and hence a greatest left annihilator. The definition of anticodomain is similar. As we are not in a boolean domain algebra, we cannot expect that the antidomain of the antidomain yields the domain or that the union of a domain element with the corresponding antidomain element equals one.

definition *ado* $x = \bigsqcup\{p.\ p \cdot x = \bot \wedge p \leq 1\}$

definition *acd* $x = \bigsqcup\{p.\ x \cdot p = \bot \wedge p \leq 1\}$

lemma *ado-acd*: *ado* $(x^\circ) =$ *acd* x
⟨*proof*⟩

lemma *acd-ado*: *acd* $(x^\circ) =$ *ado* x
⟨*proof*⟩

lemma *ado-left-zero* [*simp*]: *ado* $x \cdot x = \bot$
⟨*proof*⟩

lemma *acd-right-zero* [*simp*]: $x \cdot$ *acd* $x = \bot$
⟨*proof*⟩

lemma *ado-greatest*: $p \leq 1 \Longrightarrow p \cdot x = \bot \Longrightarrow p \leq$ *ado* x
⟨*proof*⟩

lemma *acd-greatest*: $p \leq 1 \Longrightarrow x \cdot p = \bot \Longrightarrow p \leq$ *acd* x
⟨*proof*⟩

lemma *ado-subid*: *ado* $x \leq 1$
⟨*proof*⟩

lemma *acd-subid*: *acd* $x \leq 1$
⟨*proof*⟩

lemma *ado-left-zero-iff*: $p \leq 1 \Longrightarrow (p \leq$ *ado* $x) = (p \cdot x = \bot)$
⟨*proof*⟩

lemma *acd-right-zero-iff*: $p \leq 1 \Longrightarrow (p \leq acd\ x) = (x \cdot p = \bot)$
⟨*proof*⟩

This gives an eqational characterisation of antidomain and anticodomain.

lemma *ado-cd-bot*: $ado\ x = cd\ (\bot \leftarrow x)$
⟨*proof*⟩

lemma *acd-do-bot*: $acd\ x = do\ (x \rightarrow \bot)$
⟨*proof*⟩

lemma *ado-cd-bot-id*: $ado\ x = 1 \sqcap (\bot \leftarrow x)$
⟨*proof*⟩

lemma *acd-do-bot-id*: $acd\ x = 1 \sqcap (x \rightarrow \bot)$
⟨*proof*⟩

lemma *ado-cd-bot-var*: $ado\ x = cd\ (\bot \leftarrow do\ x)$
⟨*proof*⟩

lemma *acd-do-bot-var*: $acd\ x = do\ (cd\ x \rightarrow \bot)$
⟨*proof*⟩

lemma *ado-do-bot*: $ado\ x = do\ (do\ x \rightarrow \bot)$
⟨*proof*⟩

lemma $do\ x = ado\ (ado\ x)$
⟨*proof*⟩

lemma *acd-cd-bot*: $acd\ x = cd\ (\bot \leftarrow cd\ x)$
⟨*proof*⟩

lemma *ado-do-bot-var*: $ado\ x = 1 \sqcap (do\ x \rightarrow \bot)$
⟨*proof*⟩

lemma *acd-cd-bot-var*: $acd\ x = 1 \sqcap (\bot \leftarrow cd\ x)$
⟨*proof*⟩

Domain and codomain are compatible with the boxes.

lemma *cd-box-ado*: $cd\text{-}box\ x\ \bot = ado\ x$
⟨*proof*⟩

lemma *do-box-acd*: $do\text{-}box\ x\ \bot = acd\ x$
⟨*proof*⟩

lemma *ado-subid-prop*: $p \leq 1 \Longrightarrow ado\ p = 1 \sqcap (p \rightarrow \bot)$
⟨*proof*⟩

lemma *ado-do*: $p \leq 1 \Longrightarrow ado\ p = do\ (p \rightarrow \bot)$
⟨*proof*⟩

lemma *ado-do-compl*: $ado\ x \cdot do\ x = \bot$
⟨*proof*⟩

lemma $ado\ x \sqcup do\ x = \top$
⟨*proof*⟩

lemma $\forall x\ p.\ \exists f.\ 1 \sqcap (\top \cdot p \leftarrow x) = 1 \sqcap (\bot \leftarrow (x \rightarrow p \cdot \top))$
⟨*proof*⟩

lemma $cd\text{-}box\ x\ p = ado\ (x \cdot ado\ p)$
⟨*proof*⟩

lemma *ad-do-bot* [*simp*]: $(1 \sqcap (do\ x \rightarrow \bot)) \cdot do\ x = \bot$
⟨*proof*⟩

lemma *do-heyting-galois*: $(do\ x \cdot do\ y \leq do\ z) = (do\ x \leq 1 \sqcap (do\ y \rightarrow do\ z))$
⟨*proof*⟩

lemma *do-heyting-galois-var*: $(do\ x \cdot do\ y \leq do\ z) = (do\ x \leq cd\text{-}box\ (do\ y)\ (do\ z))$
⟨*proof*⟩

Antidomain is therefore Heyting negation.

lemma *ado-heyting-negation*: $ado\ (do\ x) = cd\text{-}box\ (do\ x)\ \bot$
⟨*proof*⟩

lemma *ad-ax1* [*simp*]: $(1 \sqcap (do\ x \rightarrow \bot)) \cdot x = \bot$
⟨*proof*⟩

lemma $1 \sqcap (do\ (1 \sqcap (do\ x \rightarrow \bot)) \rightarrow \bot) = do\ x$
⟨*proof*⟩

lemma $p \leq 1 \Longrightarrow do\text{-}dia\ x\ p = 1 \sqcap (cd\text{-}box\ x\ (1 \sqcap (p \rightarrow \bot)) \rightarrow \bot)$
⟨*proof*⟩

lemma $p \leq 1 \Longrightarrow cd\text{-}box\ x\ p = 1 \sqcap (do\text{-}dia\ x\ (1 \sqcap (p \rightarrow \bot)) \rightarrow \bot)$
⟨*proof*⟩

lemma $p \leq 1 \Longrightarrow cd\text{-}dia\ x\ p = 1 \sqcap (do\text{-}box\ x\ (1 \sqcap (p \rightarrow \bot)) \rightarrow \bot)$
⟨*proof*⟩

lemma $p \leq 1 \Longrightarrow do\text{-}box\ x\ p = 1 \sqcap (cd\text{-}dia\ x\ (1 \sqcap (p \rightarrow \bot)) \rightarrow \bot)$
⟨*proof*⟩

end

6.4 Boolean Dedekind quantales

class *distributive-dedekind-quantale* = *distrib-unital-quantale* + *dedekind-quantale*

class *boolean-dedekind-quantale* = *bool-unital-quantale* + *distributive-dedekind-quantale*

begin

lemma *ad-do-bot* [*simp*]: $(1 - do\ x) \cdot do\ x = \bot$
⟨*proof*⟩

lemma *ad-ax1* [*simp*]: $(1 - do\ x) \cdot x = \bot$
⟨*proof*⟩

lemma *ad-do* [*simp*]: $1 - do\ (1 - do\ x) = do\ x$
⟨*proof*⟩

lemma *ad-ax2*: $1 - do\ (x \cdot y) \sqcup (1 - do\ (x \cdot (1 - do\ (1 - do\ y)))) = 1 - do\ (x \cdot (1 - do\ (1 - do\ y)))$
⟨*proof*⟩

lemma *ad-ax3* [*simp*]: $do\ x \sqcup (1 - do\ x) = 1$
⟨*proof*⟩

sublocale *bdad*: *antidomain-semiring* $\lambda x.\ 1 - do\ x\ (\sqcup)\ (\cdot)\ 1\ \bot$ - -
⟨*proof*⟩

sublocale *bdadka*: *antidomain-kleene-algebra* $\lambda x.\ 1 - do\ x\ (\sqcup)\ (\cdot)\ 1\ \bot$ - - *qstar*⟨*proof*⟩

sublocale *bdar*: *antirange-semiring* $(\sqcup)\ (\cdot)\ 1\ \bot\ \lambda x.\ 1 - cd\ x$ - -
⟨*proof*⟩

sublocale *bdaka*: *antirange-kleene-algebra* $(\sqcup)\ (\cdot)\ 1\ \bot$ - - *qstar* $\lambda x.\ 1 - cd\ x$⟨*proof*⟩

sublocale *bmod*: *modal-semiring-simp* $\lambda x.\ 1 - do\ x\ (\sqcup)\ (\cdot)\ 1\ \bot$ - - $\lambda x.\ 1 - cd\ x$⟨*proof*⟩

sublocale *bmod*: *modal-kleene-algebra-simp* $(\sqcup)\ (\cdot)\ 1\ \bot$ - - *qstar* $\lambda x.\ 1 - do\ x\ \lambda x.\ 1 - cd\ x$⟨*proof*⟩

lemma *inv-neg*: $(-x)^\circ = -(x^\circ)$
⟨*proof*⟩

lemma *residuation*: $x^\circ \cdot -(x \cdot y) \leq -y$
⟨*proof*⟩

lemma *bres-prop*: $x \to y = -(x^\circ \cdot -y)$
⟨*proof*⟩

lemma *fres-prop*: $x \leftarrow y = -(-x \cdot y^{\circ})$
⟨*proof*⟩

lemma *do-dia-fdia*: *do-dia x p = bdad.fdia x p*
⟨*proof*⟩

lemma *cd-dia-bdia*: *cd-dia x p = bdar.bdia x p*
⟨*proof*⟩

lemma *do-dia-fbox-de-morgan*: $p \leq 1 \Longrightarrow$ *do-dia x p* $=$ *1* $-$ *bdad.fbox x* (*1* $-$ *p*)
⟨*proof*⟩

lemma *fbox-do-dia-de-morgan*: $p \leq 1 \Longrightarrow$ *bdad.fbox x p* $=$ *1* $-$ *do-dia x* (*1* $-$ *p*)
⟨*proof*⟩

lemma *cd-dia-bbox-de-morgan*: $p \leq 1 \Longrightarrow$ *cd-dia x p* $=$ *1* $-$ *bdar.bbox x* (*1* $-$ *p*)
⟨*proof*⟩

lemma *bbox-cd-dia-de-morgan*: $p \leq 1 \Longrightarrow$ *bdar.bbox x p* $=$ *1* $-$ *cd-dia x* (*1* $-$ *p*)
⟨*proof*⟩

lemma *do-box-bbox*: $p \leq 1 \Longrightarrow$ *do-box x p = bdar.bbox x p*
⟨*proof*⟩

lemma *cd-box-fbox*: $p \leq 1 \Longrightarrow$ *cd-box x p = bdad.fbox x p*
⟨*proof*⟩

lemma *do-dia-cd-box-de-morgan*: $p \leq 1 \Longrightarrow$ *do-dia x p* $=$ *1* $-$ *cd-box x* (*1* $-$ *p*)
⟨*proof*⟩

lemma *cd-box-do-dia-de-morgan*: $p \leq 1 \Longrightarrow$ *cd-box x p* $=$ *1* $-$ *do-dia x* (*1* $-$ *p*)
⟨*proof*⟩

lemma *cd-dia-do-box-de-morgan*: $p \leq 1 \Longrightarrow$ *cd-dia x p* $=$ *1* $-$ *do-box x* (*1* $-$ *p*)
⟨*proof*⟩

lemma *do-box-cd-dia-de-morgan*: $p \leq 1 \Longrightarrow$ *do-box x p* $=$ *1* $-$ *cd-dia x* (*1* $-$ *p*)
⟨*proof*⟩

end

class *dc-involutive-modal-quantale* = *dc-modal-quantale* + *involutive-quantale*

begin

sublocale *invmqmka*: *involutive-dr-modal-kleene-algebra* (⊔) (·) *1* ⊥ (≤) (<) *qstar* *invol dom cod*⟨*proof*⟩

lemma *do-approx-dom*: *do x* $\leq$ *dom x*

⟨*proof*⟩

end

class *dc-modal-quantale-converse* = *dc-involutive-modal-quantale* + *quantale-converse*

sublocale *dc-modal-quantale-converse* ⊆ *invmqmka*: *dr-modal-kleene-algebra-converse* (⊔) (·) *1* ⊥ (≤) (<) *qstar invol dom cod*⟨*proof*⟩

class *dc-modal-quantale-strong-converse* = *dc-involutive-modal-quantale* +
assumes *weak-dom-def*: $dom\ x \leq x \cdot x^{\circ}$
and *weak-cod-def*: $cod\ x \leq x^{\circ} \cdot x$

begin

sublocale *invmqmka*: *dr-modal-kleene-algebra-strong-converse* (⊔) (·) *1* ⊥ (≤) (<) *qstar invol dom cod*
⟨*proof*⟩

lemma *dom-def*: $dom\ x = 1 \sqcap (x \cdot x^{\circ})$
⟨*proof*⟩

lemma *cod-def*: $cod\ x = 1 \sqcap (x^{\circ} \cdot x)$
⟨*proof*⟩

lemma *do-dom*: $do\ x = dom\ x$
⟨*proof*⟩

lemma *cd-cod*: $cd\ x = cod\ x$
⟨*proof*⟩

end

class *dc-modal-dedekind-quantale* = *dc-involutive-modal-quantale* + *dedekind-quantale*

class *cd-distributive-modal-dedekind-quantale* = *dc-modal-dedekind-quantale* + *distrib-unital-quantale*

class *dc-boolean-modal-dedekind-quantale* = *dc-modal-dedekind-quantale* + *bool-unital-quantale*

begin

lemma *subid-idem*: $p \leq 1 \Longrightarrow p \cdot p = p$
⟨*proof*⟩

lemma *subid-comm*: $p \leq 1 \Longrightarrow q \leq 1 \Longrightarrow p \cdot q = q \cdot p$
⟨*proof*⟩

lemma *subid-meet-comp*: $p \leq 1 \Longrightarrow q \leq 1 \Longrightarrow p \sqcap q = p \cdot q$

⟨*proof*⟩

lemma *subid-dom*: $p \leq 1 \Longrightarrow dom\ p = p$
⟨*proof*⟩

lemma *do-prop*: $(do\ x \leq do\ y) = (x \leq do\ y \cdot \top)$
⟨*proof*⟩

lemma *do-lla*: $(do\ x \leq do\ y) = (x \leq do\ y \cdot x)$
⟨*proof*⟩

lemma *lla-subid*: $p \leq 1 \Longrightarrow ((dom\ x \leq p) = (x \leq p \cdot x))$
⟨*proof*⟩

lemma *dom-do*: $dom\ x = do\ x$
⟨*proof*⟩

end

end

References

[1] A. Armstrong, G. Struth, and T. Weber. Kleene algebra. *Archive of Formal Proofs*, 2013.

[2] C. Calk, E. Goubault, P. Malbos, and G. Struth. Algebraic coherent confluence and higher globular Kleene algebras. *Logical Methods in Computer Science*, 18(4), 2022.

[3] C. Calk, P. Malbos, D. Pous, and G. Struth. Higher catoids, higher quantales and their correspondences. *arXiv*, 2307.09253, 2023.

[4] U. Fahrenberg, C. Johansen, G. Struth, and K. Ziemiański. Catoids and modal convolution algebras. *Algebra Universalis*, 84:10, 2023.

[5] V. B. F. Gomes, W. Guttmann, P. Höfner, G. Struth, and T. Weber. Kleene algebras with domain. *Archive of Formal Proofs*, 2016.

[6] V. B. F. Gomes and G. Struth. Modal Kleene algebra applied to program correctness. In *FM 2016*, volume 9995 of *LNCS*, pages 310–325, 2016.

[7] D. Pous and G. Struth. Dedekind quantaloids as intuitionistic modal algebras. In preparation, 2023.

[8] P. Resende. Open maps of involutive quantales. *Applied Categorical Structures*, 26:631–644, 2018.

[9] K. I. Rosenthal. *The Theory of Quantaloids*. Addison Wesley Longman Limited, 1996.

[10] G. Struth. Quantales. *Archive of Formal Proofs*, 2018.

[11] A. Tarski. On the calculus of relations. *Journal of Symbolic Logic*, 6(3):73–89, 1941.