(Homo|iso|epi|mono|endo|auto)-morphisms

- A **morphism** (or **homomorphism**) is an application \( f \in S_1 \mapsto S_2 \) between two sets \( S_1 \) and \( S_2 \) equipped with operations

\[
g \in S_1^n \mapsto S_1 \\
g' \in S_2^n \mapsto S_2
\]

such that \( \forall x_1, \ldots, x_n \in S_1:\)

\[
f(g(x_1, \ldots, x_n)) = g'(f(x_1), \ldots, f(x_n))
\]

- If \( n = 1 \) then \( f \circ g = g' \circ f \), diagramatically:

- an **isomorphism** is a bijective morphism
- an **epimorphism** is an onto/surjective morphism
- an **monomorphism** is a one-to-one/injective morphism
- an **endomorphism** has \( S_1 = S_2 \)
- an **automorphism** is a bijective endomorphism
- The morphism may be relative to relations \( r \subseteq S_1^2 \) and \( r' \subseteq S_2^2 \) such that for all \( \langle x_1, \ldots, x_n \rangle \in S_1^r \):

\[
\langle x_1, \ldots, x_n \rangle \in r \implies \langle f(x_1), \ldots, f(x_n) \rangle \in r'
\]

- For binary relations:

\[
x_1 r x_2 \implies f(x_1) r' f(x_2)
\]

----

Complete (homo|iso|epi|mono|endo|auto)-morphisms

- A complete morphism (or homomorphism) is an application \( f \in S_1 \mapsto S_2 \) between two sets \( S_1 \) and \( S_2 \) equipped with operations

\[
G \in \mathcal{G}(S_1) \mapsto S_1
\]

\[
G' \in \mathcal{G}(S_2) \mapsto S_2
\]

such that \( \forall X \subseteq S_1 \):

\[
f(G(X)) = G'(f(X)) \text{ where } f(X) \overset{\text{def}}{=} \{ f(x) \mid x \in X \}
\]

----

Monotone maps

- Let \( \langle P, \leq \rangle \) and \( \langle Q, \sqsubseteq \rangle \) be two posets. A map \( f \in P \mapsto Q \) is \textit{monotone} iff

\[
\forall x, y \in P : (x \leq y) \implies (f(x) \sqsubseteq f(y))
\]

- Alternatives
  - order-preserving
  - isotone
  - increasing

----

- Diagrammatically:

- if \( f \) is bijective, onto, one-to-one then \( f \) is a complete \textit{iso-}, \textit{epi-}, \textit{mono-}morphism. If \( S_1 = S_2 \) then \( f \) is a complete endomorphism, and a complete automorphism when \( f \) is bijective.
- order morphism
- ...
- Example:

- Monotony \(^1\) is self-dual (the dual of “monotone” is “monotone”)

\(^1\) Alternative “Monotonicity”;

---

**Antitone (decreasing) maps**

- Let \( \langle P, \leq \rangle \) and \( \langle Q, \sqsubseteq \rangle \) be two posets. A map \( f \in P \mapsto Q \) is **antitone** iff
  \[
  \forall x, y \in P : (x \leq y) \implies (f(x) \sqsubseteq f(y))
  \]

- Alternatives
  - order-inversing
  - decreasing
  - ...

- Self-dual notion

---

**Characterization of monotone maps using lubs**

**THEOREM.** Let \( \langle P, \leq \rangle \) and \( \langle Q, \sqsubseteq \rangle \) be two posets and \( f \in P \mapsto Q \). If \( f \) is monotone then whenever \( S \subseteq P \) and both lubs \( \sqcap S \) exists in \( P \) and \( \sqcup f(S) \) exists in \( Q \) then:

\[
\sqcup f(S) \sqsubseteq f(\sqcap S)
\]

The reciprocal is false but holds for join-semi-lattices.

**PROOF.** Assume \( f \) is monotone, \( \sqcap S \) and \( \sqcup f(S) \) exist. Then \( \forall s \in S : s \leq \sqcap S \) so by monotony \( f(s) \leq f(\sqcap S) \) whence \( \sqcup f(S) \sqsubseteq f(\sqcap S) \) by def. lub.
The inclusion can be strict, as shown by the following example:

- **$f$ is monotone**
- \[ \bigwedge f\{a, b\} = f(a) \leq f(b) = x \cap x = x \]
- \[ \sqcap z = f(c) = f(a \lor b) \]

**Characterization of monotone maps using glbs**

**Theorem.** Let $\langle P, \leq \rangle$ and $\langle Q, \sqsubseteq \rangle$ be two posets and $f \in P \mapsto Q$. If $f$ is monotone then whenever $S \subseteq P$, the glbs $\bigwedge S$ exists in $P$ and $\sqcap f(S)$ exists in $Q$, we have:

\[ \sqcap f(S) \equiv f(\bigwedge S). \]

The reciprocal is false but holds for meet-semi-lattices.

**Proof.** By duality.

---

**Order embedding**

- Let $\langle P, \leq \rangle$ and $\langle Q, \sqsubseteq \rangle$ be two posets. A map $f \in P \mapsto Q$ is an **order embedding** (written $f \in P \hookrightarrow Q$ or $f \in P \mapsto Q$) iff \[
\forall x, y \in P : x \leq y \iff f(x) \sqsubseteq f(y)
\]

- Example:

---

**An order embedding is injective**

**Theorem.** Let $\langle P, \leq \rangle$ and $\langle Q, \sqsubseteq \rangle$ be two posets and $f \in P \mapsto Q$ be an order-embedding. $f$ is injective.

**Proof.**

- $f(x) = f(y)$
- $\implies f(x) \sqsubseteq f(y) \cap f(y) \sqsubseteq f(x)$
- $\implies x \leq y \land y \leq x$
- $\implies x = y$ and so
- $x \neq y \implies f(x), \neq f(y)$
Order isomorphism

- Let $\langle P, \leq \rangle$ and $\langle Q, \sqsubseteq \rangle$ be posets. An order-iso-
  morphism is an order-embedding which is onto (whence bi-
  jective).

- Example:

\begin{center}
\includegraphics[width=0.5\textwidth]{example_diagram.png}
\end{center}

- Let $\langle P, \leq \rangle$ and $\langle Q, \sqsubseteq \rangle$ be posets. These ordered or-
  dered sets are therefore order-isomorphic if and only if

$$\exists \varphi \in P \mapsto Q : \exists \psi \in Q \mapsto P :$$

- $\varphi \circ \psi = 1_Q$
- $\psi \circ \varphi = 1_P$
- $\varphi$ is monotone
- $\psi$ is monotone

\[1_{S}\] is the identity map on set $S$.

Example of order isomorphism:

boolean encoding of finite sets

**Theorem.** Let $X = \{x_1, x_2, \ldots, x_n\}$ be a finite set. De-
fine

$$\varphi : \wp(X) \mapsto 2^n$$

$$\varphi(S) \overset{\text{def}}{=} \lambda i. (x_i \in S \equiv t t \lor f f)$$

The $\varphi$ is an order-isomorphism between $\langle \wp(X), \subseteq \rangle$ and

$\langle 2^n, \leq \rangle$ where $\leq$ is the componentwise ordering based

on $ff \leq ff < tt \leq tt$.

**Proof.**

- $x \subseteq Y$

$$\iff \forall i \in [1, n] : x_i \in X \implies x_i \in Y$$

$$\iff \forall i \in [1, n] : \varphi(x)_i \leq \varphi(Y)_i$$

$$\iff \varphi(X) \leq \varphi(Y) \text{ on } 2^n$$

If $X \neq Y$ then there is a $x_i \in X$ not in $Y$ (or inversely) so

$\varphi(x)_i = tt$ and

$\varphi(Y)_i = ff$ (or inversely), proving that $\varphi(X) \neq \varphi(Y)$ hence $\varphi$ is injective.

- Given $(b_1, \ldots, b_n) \in 2^n$, we take $S = \{x_i \in S | b_i = tt\}$ so that $\varphi(S) = (b_1, \ldots, b_n)$ proving that $\varphi$ is onto.

Used to encode finite sets as bit vectors.
Embedding of a poset in its powerset

**Theorem.** Let \((P, \leq)\) be a poset. Then there is a set \(Q \subseteq \wp(P)\) of subsets of \(P\) such that \((P, \leq)\) is order-isomorphic to \((Q, \subseteq)\).

**Proof.**
- Define \(Q = \{\{x\} | x \in P\}\)
- Define \(\varphi : P \to Q\) by \(\varphi(x) = \{x\}\)
- \(\varphi\) is a bijection
- \((x \leq y) \iff (\{x\} \subseteq \{y\})\)

**Example:**

Join/meet preserving maps

- let \((P, \leq)\) and \((Q, \sqsubseteq)\) be two posets. The map \(f \in P \to Q\) is called join preserving whenever if \(x, y \in P\) and the lub \(x \lor y\) exists in \(P\) then the lub \(f(x) \sqsubseteq f(y)\) does exist in \(Q\) and is such that:
  \[ f(x \lor y) = f(x) \sqsubseteq f(y) \]
- Example:

- \(f(c \lor d) = f(c) = e = y \sqsubseteq z = f(c) \sqsubseteq f(d)\)
- \(b \lor c\) does not exist so there is no requirement on \(f(b) \sqsubseteq f(c)\)

Join/meet preserving maps are monotone

**Theorem.** A join or meet preserving map is monotone.

**Proof.**
- if \(x \leq y\) then \(x \sqsubseteq y = y\) does exists. So \(f(x) \sqsubseteq f(y)\) hence \(f(x) \sqsubseteq f(y)\) since \(f\) preserves existing, proving that \(f(x) \subseteq f(y)\) by def. of lbs.
- By duality a meet-preserving maps is monotone (since the dual of monotone is monotone)
Not all monotone maps preserve lubs/glbs

Counter-example:

- \( f \) is monotone
- \( f(x \lor y) = f(z) = b \)
- \( f(x) \perp f(y) = a \perp a = a \neq b \)

Complete join preserving maps

- Let \( \langle P, \leq \rangle \) and \( \langle Q, \sqsubseteq \rangle \) be two posets. The map \( f \in P \to Q \) is a complete join preserving whenever it preserves existing lubs:

\[
\forall X \subseteq P : \lor X \text{ exists } \implies f(\lor X) = \lor f(X)
\]

- The dual notion is that of complete meet preserving map:

\[
\forall X \subseteq P : \land X \text{ exists } \implies f(\land X) = \land f(X)
\]

Not all finite join/meet preserving maps are complete

- Example of finite join preserving map which is not a complete join preserving map:

- \( \varphi \) is not a complete join morphism:

\[
\varphi(\bigcup \omega) = \varphi(\bigcup \{0, 1, 2, \ldots\}) = \varphi(\omega) = b \neq a = \bigcup \{a\} = \bigcup \{\varphi(x) \mid x \in \omega\} = \bigcup \varphi(\omega)
\]

- \( \varphi \) is a join morphism
- \( \psi \) is a complete join morphism
Continuous and co-continuous maps

- A map $f : P \mapsto Q$ from a poset $(P, \leq)$ into a poset $(Q, \sqsubseteq)$ is **continuous** (or **upper-continuous**) if and only if for all chains $C$ of $P$ such that $\bigvee C$ exists then $\bigcup f(C)$ exists and we have
  $$f(\bigvee C) = \bigcup f(C)$$

- Often this hypothesis is needed only for denumerable chains.
  $f$ is $\omega$-continuous iff for all increasing chains $x_0 \leq x_1 \leq \ldots \leq x_n \leq \ldots$ of $P$ such that $\bigvee_{i \in \mathbb{N}} x_i$ exists then $\bigcup_{i \geq 0} f(x_i)$ exists and
  $$f(\bigvee_{i \in \mathbb{N}} x_i) = \bigcup_{i \in \mathbb{N}} f(x_i)$$

---

**Theorem.** Let $f : P \mapsto Q$, $(P, \leq)$ be a poset. If $f$ is $\omega$-continuous (preserves exists lubs of denumerable chains) then $f$ is monotone.

**Proof.** If $x \leq y$ the denumerable chain $x \leq y \leq y \leq \ldots$ has a lub $y$, so by $\omega$-continuity of $f$, $f(y) = f(\bigvee\{x, y\}) = f(x) \vee f(y)$ proving $f(x) \leq f(y)$ by def. of lubs.

- By duality, $\omega$-co-continuous maps are monotone

---

**Example ($\varphi$) and counter-example ($\psi$):**

---

- The reciprocal is not true. A monotone map may not be $\omega$-continuous, as shown by the following counter-example:

  - $f(x) = x + 1, \ x \leq \omega$
  - $f(\omega + 1) = \omega + 1$
  - $f$ is monotone
  - $f$ is not continuous since
    $$f(\bigcup_{n<\omega} n) = f(\omega) = \omega + 1$$
    $$\bigcup_{n<\omega} f(n) = \bigcup_{n<\omega} (n + 1) = \omega = \omega$$
Chain conditions and continuity

**Theorem.** Let \( \langle P, \leq \rangle \) be a poset satisfying the ascending chain condition (ACC) and \( \langle Q, \sqsubseteq \rangle \) be a poset. Then any monotone map \( f \in P \rightarrow Q \) is continuous.

**Proof.** Let \( \{ x_\delta, \delta \in \Omega \} \) be an increasing chain of elements of \( P \). By the ACC, \( \exists k < \omega : \forall \delta \geq k : x_\delta = x_k \) so that \( \bigvee_{\delta \in \Omega} x_\delta = x_k \). It follows that \( f(\bigvee_{\delta \in \Omega} x_\delta) = f(x_k) \). Since \( \forall \delta \in \Omega : x_\delta \leq x_k \) and \( f \) is monotone, we have \( f(x_\delta) \leq f(x_k) \) whence \( \bigvee_{\delta \in \Omega} f(x_\delta) \leq f(x_k) \). Put \( f(x_k) \in \{ f(x_\delta) \mid \delta \in \Omega \} \) so \( f(x_k) \leq \bigvee_{\delta \in \Omega} f(x_\delta) \) and by antisymmetry \( \bigvee_{\delta \in \Omega} f(x_\delta) = f(x_k) \). It follows that \( \bigvee_{\delta \in \Omega} f(x_\delta) = f(x_k) = f(\bigvee_{\delta \in \Omega} x_\delta) \), proving continuity.

By duality, if \( \langle P, \leq \rangle \) is a poset satisfying the descending chain condition (DCC) and \( \langle Q, \sqsupseteq \rangle \) is a poset then any monotone map \( f \in P \rightarrow Q \) is co-continuous.

---

Boolean lattice morphism

- Let \( \langle P, \lor, \land \rangle \) and \( \langle Q, \bot, \top \rangle \) be lattices. A **lattice morphism** \( f \in P \rightarrow Q \) satisfies:

\[
\begin{align*}
f(x \lor y) &= f(x) \land f(y) \\
f(x \land y) &= f(x) \lor f(y)
\end{align*}
\]

- Let \( \langle P, 0, 1, \lor, \land \rangle \) and \( \langle Q, 0, 1, \lor, \land \rangle \) be boolean algebras.

A **Boolean algebra morphism** \( f \in P \rightarrow Q \) if and only if:

- \( f \) is a lattice morphism
- \( f(0) = 0 \)
- \( f(1) = 1 \)
- \( f(-x) = f(x)' \)

---

On the conditions defining the Boolean lattice morphisms

**Theorem.** Let \( \langle P, 0, 1, \lor, \land \rangle \) and \( \langle Q, 0, 1, \lor, \land \rangle \) be boolean algebras. Assume \( f \) is a lattice morphism.

(i) \( (a) \ f(0) = 0 \) and \( f(1) = 1 \)

\[ \iff (b) \ f(-a) = (f(a))' \quad \forall a \in P \]

(ii) If \( f(-a) = (f(a))' \), then

\[ \begin{align*}
& (c) \ f(a \lor b) = f(a) \lor f(b) \\
& \iff (d) \ f(a \land b) = f(a) \land f(b)
\end{align*} \]
Notations for monotone, lub/glb preserving and (co-)continuous maps

Let $\langle P, \leq \rangle$ and $\langle Q, \sqsubseteq \rangle$ be posets. We define:

$\langle P, \leq \rangle \twoheadrightarrow \langle Q, \sqsubseteq \rangle$ (or $P \twoheadrightarrow Q$ if $\leq$ and $\sqsubseteq$ are understood) to be the set of monotone maps of $P$ into $Q$.

$\langle P, \leq \rangle \hookrightarrow \langle Q, \sqsubseteq \rangle$ (or $P \hookrightarrow Q$ if $\leq$ and $\sqsubseteq$ are understood) to be the set of complete lub-preserving maps of $P$ into $Q$.

$\langle P, \leq \rangle \hookrightarrow \langle Q, \sqsubseteq \rangle$ (or $P \hookrightarrow Q$ if $\leq$ and $\sqsubseteq$ are understood) to be the set of complete glb-preserving maps of $P$ into $Q$.

The complete lattice of pointwise ordered maps on a complete lattice

**Theorem.** Let $P$ be a set and $\langle Q, \sqsubseteq, \bot, \top, \sqcap, \sqcup \rangle$ be a complete lattice. Let $\sqsubseteq$ be the pointwise ordering of maps $f \in P \mapsto L$: $f \sqsubseteq g \iff \forall x \in P : f(x) \sqsubseteq g(x)$. Then $\langle P \mapsto Q, \sqsubseteq, \bot, \top, \sqcap, \sqcup \rangle$ (where $\bot \overset{\text{def}}{=} \lambda x. \bot$, $\top \overset{\text{def}}{=} \lambda x. \top$, $\sqcup F \overset{\text{def}}{=} \lambda x. \sqcup_{f \in F} f(x)$ and $\sqcap F \overset{\text{def}}{=} \lambda x. \sqcap_{f \in F} f(x)$) is a complete lattice.  

\[ \square \]
The complete lattice of pointwise ordered monotone maps on a complete lattice

**Theorem.** Let \(\langle P, \leq \rangle\) be a poset and \(\langle Q, \sqsubseteq, \bot, \top, \lor, \land \rangle\) be a complete lattice. The set of monotonic maps of \(P\) into \(Q\) is a complete lattice \(\langle P \mapsto Q, \sqsubseteq, \bot, \top, \lor, \land \rangle\).

**Proof.** 
1. The ordering \(\sqsubseteq\) makes \(\langle P \mapsto Q, \sqsubseteq\rangle\) a complete lattice.
   - Since \(\langle P, \leq \rangle \subseteq \langle P \mapsto Q, \sqsubseteq \rangle\), it follows that \(\langle P \mapsto Q, \sqsubseteq \rangle\) is a poset.
   - The lub in \(\langle P \mapsto Q, \sqsubseteq \rangle\) is \(\sqcup\) such that \(\sqcup f(x) = \sqcup_{x \in A} f(x)\).
   - Observe that \(\sqcup f_i\) is monotone since \(x \leq y\) implies \(f_i(x) \sqsubseteq f_i(y)\) whenever \(x \in A\) and \(f_i \in P \mapsto Q\).

2. It follows that \(\sqcup f_i\) is also the lub in \(P \mapsto Q\).

\(\square\)
Encoding Maps between Posets

- Since \( P \rightarrow Q \) has \( \text{triv} \), it also has \( \text{glb} \) which may not coincide with the pointwise \( \text{glb} \) in \( \langle P \rightarrow Q, \text{\textcap} \rangle \), as shown by the following counterexample:

![Counterexample diagram]

Reference
Boolean terms

- Let $\langle B, 0, 1, \lor, \land, \neg \rangle$ be a boolean algebra
- Let $\mathcal{V}$ be a set of variables and $\langle x_1, \ldots, x_n \rangle \in \mathcal{V}^n$
- The boolean terms $\text{Bt}(B; \langle x_1, \ldots, x_n \rangle)$ are defined by the following grammar:

$$T ::= x_i \mid 0 \mid 1 \mid T_1 \lor T_2 \mid T_1 \land T_2 \mid \neg T_1 \mid (T_1)$$

The interpretation of Boolean terms

- The semantics or interpretation $S[T] \in 2^n \mapsto 2$ of $T \in \text{Bt}(B; \langle x_1, \ldots, x_n \rangle)$ is defined by

$$S[x_i](v_1, \ldots, v_n) \equiv v_i$$
$$S[0](v_1, \ldots, v_n) \equiv 0$$
$$S[1](v_1, \ldots, v_n) \equiv 1$$
$$S[T_1 \lor T_2](v_1, \ldots, v_n) \equiv S[T_1](v_1, \ldots, v_n) \lor S[T_2](v_1, \ldots, v_n)$$
$$S[T_1 \land T_2](v_1, \ldots, v_n) \equiv S[T_1](v_1, \ldots, v_n) \land S[T_2](v_1, \ldots, v_n)$$
$$S[\neg T](v_1, \ldots, v_n) \equiv \neg S[T](v_1, \ldots, v_n)$$
$$S[(T_1)](v_1, \ldots, v_n) \equiv S[T_1](v_1, \ldots, v_n)$$

Encoding of Boolean functions by Boolean terms

- The encoding of $v = \langle v_1, \ldots, v_n \rangle \in 2^n$ over variables $\langle x_1, \ldots, x_n \rangle$ is:

$$\text{Te}(v)(x_1, \ldots, x_n) = (v_1 = 1 ? x_1 : \neg x_1) \land \ldots \land (v_n = 1 ? x_n : \neg x_n)$$

- The encoding of $f \in 2^n \mapsto 2$ over variables $\langle x_1, \ldots, x_n \rangle$ is:

$$\text{Te}(f)(x_1, \ldots, x_n) = \bigvee \{\text{Te}(v)(x_1, \ldots, x_n) \mid v \in 2^n \land f(v) = 1\}$$

Theorem. For all $a = \langle a_1, \ldots, a_n \rangle \in 2^n$ and $b = \langle b_1, \ldots, b_n \rangle \in 2^n$:

$$S[\text{Te}(a)(x_1, \ldots, x_n)]b = 1 \quad \text{iff} \quad b = a \quad = 0 \quad \text{iff} \quad b \neq a$$

Proof.

$S[\text{Te}(a)(x_1, \ldots, x_n)]b$
$= (a_1 = 1 ? S[x_1]b : \neg S[x_1]b) \land \ldots \land (a_n = 1 \land S[x_n]b \land \neg S[x_n]b)$
$= (a_1 = 1 ? b_1 : \neg b_1) \land \ldots \land (a_n = 1 \land b_n : \neg b_n)$
$= (a_1 = b_1 \land \ldots \land a_n = b_n)$
$= a = b$
$\begin{cases} 1 & \text{iff } a = b \\ 0 & \text{iff } a \neq b \end{cases}$
Theorem. $2^n \rightarrow 2$ and $\{\text{Te}(f)(x_1, \ldots, x_n) \mid f \in 2^n \rightarrow 2\}$ are isomorphic by $(S, \text{Te})$.

Proof.

$S[\text{Te}(f)(x_1, \ldots, x_n)]b$ where $b = \langle b_1, \ldots, b_n \rangle$

$= \bigvee \{S[\text{Te}(v)(x_1, \ldots, x_n)] | f(v) = 1\}$

$= \bigvee \{\{b = u \neq 1 \cdot 0 \mid f(u) = 1\}$

$= f(b)$

$= f(b)$

Let $T \in \{\text{Te}(f)(x_1, \ldots, x_n) \mid f \in 2^n \rightarrow 2\}$. We must show that $\text{Te}(S[T]) = T$. Given $f \in 2^n \rightarrow 2$, we have $\text{Te}(S[\text{Te}(f)(x_1, \ldots, x_n)]) = \text{Te}(f)$, Q.E.D.

Boolean terms in disjunctive normal forms

- A Boolean term over $\{x_1, \ldots, x_n\}$ is in disjunctive normal form (DNF) iff it is in the form

$$\bigvee_{i=1}^{k} \bigwedge_{j=1}^{n} \ell_{ij}$$

where $\ell_{ij}$ is $x_j$ or $-x_j$.

- Any boolean term $T$ can be put in equivalent DNF.

---

Example (conditional)

$$f(x, y, z) = (x \oplus y \oplus z)$$

$= (x \land y) \lor (-x \land z)$

$= ((-x \land z) \land (y \lor -y)) \lor ((-x \land y) \land (z \lor -z))$

$= (-x \land -y \land z) \lor (-x \land y \land z) \lor (x \land y \land -z) \lor (x \land y \land z)$

in so called "disjunctive normal form".
Encoding of Boolean functions by BDDs

Example of Shannon trees

A BDD (Binary Decision Diagram) discovered by Randal Bryant in 1986 is a compact representation of a Shannon tree of a boolean expression.

Example:

- \( f(x, y, z) = (x \land y) \lor (y \land \neg z) \lor (z \lor \neg y) \)
- Table representation:

<table>
<thead>
<tr>
<th>x</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>z</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>f</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>


- Shannon tree representation (with \( x < y < z \))

(1) Sharing: merge redundant subtrees (to get a Directed Acyclic Graph — DAG)
Shannon decomposition of Boolean functions

- Let \( \langle \text{Var}, <^V \rangle \) be a totally strictly ordered set of variables.
- Let \( \text{Var}_n = \{ V \subseteq \text{Var} \mid |V| = n \} \) be the set of \( n \) variables \( \{x_1, \ldots, x_n\} \) where, by convention, \( x_1 <^V \ldots <^V x_n \).
- Let \( B_n = \text{Var}_n \times \{\{0, 1\}^n \rightarrow \{0, 1\}\} \) be the set of pairs \( \langle x_1, \ldots, x_n; f \rangle \) denoted \( f(x_1, \ldots, x_n) \) which value at point \( x_1 = b_1; \ldots; x_n = b_n \) is \( f(b_1, \ldots, b_n) \).
- Let \( V(f(x_1, \ldots, x_n)) = \{x_1, \ldots, x_n\} \) where \( x_1 <^V \ldots <^V x_n \).

Shannon tree

- A Shannon tree over variables \( x_1 <^V \ldots <^V x_n \) is:
  - if \( n = 0 \) then 1 or \( \bot \)
  - if \( n > 0 \) then \( \langle x_1, t_1, t_2 \rangle \) where \( t_1, t_2 \) are Shannon trees over \( x_2 <^V \ldots <^V x_n \).
- Example \( x_1 = x <^V x_2 = y \)

\[\langle x, \langle y, 1, 0 \rangle; \langle y, 1, 1 \rangle \rangle\]
Isomorphism between Shannon trees and Boolean functions

- A Shannon tree $t$ over variables $x_1 <^* \ldots <^* x_n$ represents a Boolean function $f(t)(x_1, \ldots, x_n) = \text{match } t$ with
  
  \[
  \begin{align*}
  &0 \rightarrow t \quad \text{case } n = 0 \\
  &(x_1, t_1, t_2) \rightarrow (x_1 \land f(t_1)(x_2, \ldots, x_n)) \\
  &\lor (-x_1 \land f(t_2)(x_2, \ldots, x_n))
  \end{align*}
  \]

- The Shannon tree representing a Boolean function $f(x_1, \ldots, x_n)$ with $x_1 <^* \ldots <^* x_n$ is:

  \[
  \begin{align*}
  \text{Sh}(f(x_1, \ldots, x_n)) = (n = 0 ? f() : \\
  (x_1, \text{Sh}(\lambda x_2, \ldots, x_n.f(0, x_2, \ldots, x_n)), \\
  \text{Sh}(\lambda x_2, \ldots, x_n.f(1, x_2, \ldots, x_n)))
  \end{align*}
  \]

Definition of Boolean Decision Diagrams (BDD)

The BDDs are recursively defined as follows:

- 0 is a BDD
- 1 is a BDD
- if $b_1$, $b_2$ are BDDs, $x \in \text{Var}$ is a variable then $b = \langle x, b_1, b_2 \rangle$ is a BDD (with $\text{var}(b) = x$, $\text{left}(b) = b_1$, $\text{right}(b) = b_2$)

Example:

\[
\begin{array}{c|c|c|c}
  x & 0 & C & 1 \\
  \hline
  y & 0 & 1 & 0 \\
  \hline
  f(x, y) & 1 & C & 1 \\
\end{array}
\]

\[
\langle x, \langle y, 1, 0 \rangle, \langle y, 1, 1 \rangle \rangle
\]

Example:

\[
\begin{align*}
\text{b}_0 &= \text{C} \\
\text{b}_1 &= 1 \\
\text{b}_2 &= \langle z, \text{b}_1, \text{b}_0 \rangle \\
\text{b}_3 &= \langle z, \text{b}_0, \text{b}_1 \rangle \\
\text{b}_4 &= \langle y, \text{b}_3, \text{b}_2 \rangle \\
&= \langle y, \langle z, \text{C}, 1 \rangle, \langle z, 1, 0 \rangle \rangle
\end{align*}
\]
Ordered Boolean Decision Diagram (OBDD)
- Let \( \langle \text{Var}, \prec, \rangle \) be a totally strictly ordered set of variables
- A BDD \( t \) is ordered \( \text{ordered}(b) = \top \) if and only if either \( b \in \{0, 1\} \) or
  - If \( \text{left}(b) \not\in \{0, 1\} \) then \( \text{var}(b) \prec u \cdot \text{var}(\text{left}(b)) \)
  - If \( \text{right}(b) \not\in \{0, 1\} \) then \( \text{var}(b) \prec u \cdot \text{var}(\text{right}(b)) \)
  - \( \text{left}(b) \neq \text{right}(b) \)
- Counter-examples:

![Counter-examples diagram](image)

Example:

<table>
<thead>
<tr>
<th>( x )</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>( y )</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>( z )</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>( t )</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

\[ \text{sh}(x) = \begin{cases} 1 & \text{if } x = 1 \\ 0 & \text{if } x = 0 \end{cases} \]

\[ \text{obdd}(\text{sh}(x)) = \begin{cases} \top & \text{if } x = 1 \\ \bot & \text{if } x = 0 \end{cases} \]

Representation of a Shannon tree by an Ordered Boolean Decision Diagram (OBDD)
- The OBDD \( \text{obdd}(t) \) representing a Shannon tree \( t \) is defined as follows

\[
\text{obdd}(t) = \text{match } t \text{ with } \\
\quad \begin{cases} 0 | 1 \rightarrow t \\ \langle x, t_1, t_2 \rangle \rightarrow (t_1 = t_2 \oplus \text{obdd}(t_1) \oplus (x, \text{obdd}(t_1), \text{obdd}(t_2))) \end{cases}
\]

Boolean functions represented by an Ordered Boolean Decision Diagram (OBDD)
- An OBDD no longer represents one function \( f \) but rather all functions whose results are the same regardless of the assignment of additional variables absent in the BDD
- Example: If \( \forall x, y, z : f(x, y, z) = g(y) \) then

\[ \text{obdd}(\text{sh}(f(x, y, z))) = \text{obdd}(\text{sh}(g(y))) \]

For example if \( g(y) = \neg y \) then this OBDD is

![Example diagram](image)
- If this does not matter, then it is sufficient to memorize the CBDD as well as the corresponding set of variables \( \{x, y, z\} \) or \( \{y\} \) in the above example.

**Typed Shannon tree**

- The idea of *typed Shannon tree* [2] came from the remark that

\[ -f = (-x \land -f_x) \lor (-x \land -f_z) \]

so that the Shannon trees \( \text{Sh}(f) \) and \( \text{Sh}(-f) \) of \( f \) and \( -f \) are identical except at the leaves where \( 0 \) and \( 1 \) are exchanged

- So one can use \( +\text{Sh}(f) \) for \( \text{Sh}(f) \) and \( -\text{Sh}(f) \) for \( \text{Sh}(-f) \) with \( +1 = 1 \) and \( -1 = 0 \).

---

**Reference**


---

**Example (+ is omitted)**

- Formally a *typed Shannon tree* \( t \) over \( x_1 <^1 \ldots <^1 x_n \) is either
  - a leave 1 when \( n = 0 \), or
  - a node \( (x, (s_1, t_1), (s_2, t_2)) \) where \( s_1, s_2 \in \{-, +\} \) and \( t_1, t_2 \)
    are typed Shannon trees over \( x_1 <^1 \ldots <^1 x_n \)

---

**Boolean functions represented by a Typed Shannon tree**

- The Boolean function \( \text{bf}(t) \) represented by a typed Shannon tree \( t \) over \( x_1 <^1 \ldots <^1 x_n \) is:
  
  \[ \text{bf}(t) = \text{match } t \text{ with } \]
  
  \[ 0|1 \rightarrow \lambda(). t \text{ — case } n = 0 \]
  
  \[ (x, (s_1, t_1), (s_2, t_2)) \rightarrow \]
  
  let \( f_1(x_2, \ldots, x_n) = \text{bf}(t_1) \)
  
  and \( f_2(x_2, \ldots, x_n) = \text{bf}(t_2) \) in
  
  \[ \lambda x_1, \ldots, x_n. ~ (x_1 \land \text{bo}(s_1)(f_1(x_2, \ldots, x_n))) \]
  
  \[ \lor (-x_1 \land \text{bo}(s_2)(f_2(x_2, \ldots, x_n))) \]

  where \( \text{bo}(+)(b) = b \) while \( \text{bo}(-)(b) = -b \).
Typed Shannon trees representing a Boolean function

Let \( f(x_1, \ldots, x_n) \in B_n \) be a Boolean function over the variables \( x_1 <^* \cdots <^* x_n \). The typed Shannon tree encoding \( f \) is:

\[
\text{tsh}(f(x_1, \ldots, x_n)) = \\
( n = 1 ? (x_1 tsh(f(C) ? (f(0) ? (+, 1) : (-, 1)), \\
( f(1) ? (+, 1) : (-, 1)) \\
\) let \( \langle s_1, t_1 \rangle = ( f(0, 1, \ldots, 1) = 1 ? \\
(+, \text{tsh}(\lambda x_2 \ldots, x_n. f(0, x_2, \ldots, x_n))) \\
( -, \text{tsh}(\lambda x_2 \ldots, x_n. \neg f(0, x_2, \ldots, x_n))) \\
and \langle s_2, t_2 \rangle = ( f(1, 1, \ldots, 1) = 1 ? \\
(+, \text{tsh}(\lambda x_2 \ldots, x_n. f(0, x_2, \ldots, x_n))) \\
( -, \text{tsh}(\lambda x_2 \ldots, x_n. \neg f(0, x_2, \ldots, x_n))) \\
in \langle x_1, \langle s_1, t_1 \rangle, \langle s_2, t_2 \rangle \rangle)
\]

Examples:
- \( \text{tsh}(\lambda y. (0 = -y)) = \langle y, (-, 1), (+, 1) \rangle \)
- \( \text{tsh}(\lambda y. (0 = -y)) = \langle y, (-, 1), (+, 1) \rangle \)
- \( \text{tsh}(\lambda x, y. (x = -y)) = \langle x, (+, \langle y, (-, 1), (+, 1) \rangle), (-, \langle y, (-, 1), (+, 1) \rangle) \rangle \)

which can be represented by the following TDG.

Encoding of a Typed Shannon tree by a Typed Decision Graph (TDG)

If \( t \) is a typed Shannon tree, the corresponding TDG is obtained by applying the previous sharing and elimination rules:

\[
\text{tdg}(t) = (t = \langle s, 1 \rangle ? \langle s, 1 \rangle \\
\| t = \langle x, \langle s_1, t_1 \rangle, \langle s_2, t_2 \rangle \rangle ? \\
( (s_1 = s_2 \land t_1 = t_2) ? (s_1 = + t_1 = -t_1) \\
\| \langle x, \langle s_1, \text{tdg}(t_1) \rangle, \langle s_2, \text{tdg}(t_2) \rangle \rangle)
\]
**Example 1:** \( f(x, y, z) = (x \land y) \lor (y \land \neg z) \lor (z \land \neg y) \)

**Example 2:** \( f(x, y, z) = (y \land x) \lor (x \land \neg z) \lor (z \land \neg x) \)

The size of TDGs, although very sensitive to the variable order, is often reasonable but can be exponential in the number of variables.

---

**Boolean functions represented by a Typed Decision Graph (TDG)**

The Boolean function \( \text{bf}(t) \) represented by a TDG \( t \) over variables \( x_1, \ldots, x_n \) is defined by:

\[
\text{bf}(t)(x_1, \ldots, x_n) = \text{match } t \text{ with } \\
\begin{array}{l}
1 \rightarrow 1 \\
\langle x, \langle s_1, t_1 \rangle, \langle s_2, t_2 \rangle \rangle \rightarrow \\
\quad \langle x = x_1 ? \text{ let } f_1(x_2, \ldots, x_n) = \text{bf}(t_1)(x_2, \ldots, x_n) \rangle \\
\quad \text{and } f_2(x_2, \ldots, x_n) = \text{bf}(t_2)(x_2, \ldots, x_n) \\
\quad \text{in } (x_1 \land \text{bo}(s_1)(f_1(x_2, \ldots, x_n))) \\
\quad \lor (\neg x_1 \land \text{bo}(s_2)(f_2(x_2, \ldots, x_n))) \\
\end{array}
\]

where \( \text{bo}(+)(b) = b \) and \( \text{bo}(-)(b) = \neg b, b \in \{0, 1\} \)

**Example:**

\[
\text{bf}(\langle y, \langle +, 1 \rangle, \langle -, 1 \rangle \rangle)(y, z)
\]

\[
= (y \land \text{bo}(+)(\text{bf}(1)(z))) \lor \\
(\neg y \land \text{bo}(-)(\text{bf}(1)(z)))
\]

\[
= (y \land 1) \lor (\neg y \land \neg 1) = y
\]

\[
\text{bf}(\langle z, \langle -, 1 \rangle, \langle +, 1 \rangle \rangle)(y, z)
\]

\[
= \text{bf}(\langle z, \langle -, 1 \rangle, \langle +, 1 \rangle \rangle)(z)
\]

\[
= (z \land \text{bo}(+)(\text{bf}(1)(z))) \lor (\neg z \land \text{bo}(+)(\text{bf}(1)(z)))
\]

\[
= (z \land 1) \lor (\neg z \land \neg 1) = -z
\]

\[
\text{bf}(\langle x, \langle +, t_1 \rangle, \langle +, t_2 \rangle \rangle)(x, y, z) \text{ where } t_1 = \langle y, \langle +, 1 \rangle, \langle -, 1 \rangle \rangle \text{ and } t_2 = \langle z, \langle -, 1 \rangle, \langle +, 1 \rangle \rangle
\]

\[
= ((x \land \text{bo}(+)(\text{bf}(t_1)(y, z))) \lor (\neg x \land \text{bo}(+)(\text{bf}(t_2)(y, z))))
\]

\[
= ((x \land \text{bf}(t_1)(y, z)) \lor (\neg x \land \text{bf}(t_2)(y, z)))
\]

\[
= (x \land y) \lor (\neg x \lor \neg z)
\]
Operations on Typed Decision Graphs (TDG)

- Since the representation of a Boolean function by a TDG is unique, equality of Boolean functions can be represented by the equality (of the physical addresses) of the representations.

- Negation just inverts the signs at the leaves

\[
\neg t(x_1, \ldots, x_n) = \text{match } t \text{ with } \begin{cases} 
- \text{case } n \geq 1 \\
\langle x_1, (s_1, 1), (s_2, 1) \rangle \rightarrow \langle x_1, (-s_1, 1), (-s_2, 1) \rangle \\
\langle x_1, (s_1, 1), (s_2, t_2) \rangle \rightarrow \langle x_1, (-s_1, 1), (s_2, \neg t_2) \rangle \\
\langle x_1, (s_1, t_1), (s_2, 1) \rangle \rightarrow \langle x_1, (s_1, -t_1), (-s_2, 1) \rangle \\
\langle x_1, (s_1, t_1), (s_2, t_2) \rangle \rightarrow \langle x_1, (s_1, -t_1), (s_2, \neg t_2) \rangle 
\end{cases}
\]

where \(\neg (+) = -\) and \(\neg (-) = +\)
- Counter-examples:
  The lattice of open subsets of \( \mathbb{R} \) (that is subsets which are unions of open intervals \([a, b)\)) has no join-irreducible element.
- When the second condition is generalized to arbitrary joins \( \bigvee_{i \in \Delta} a_i \) \( z \) is called completely join-irreducible.
- In a lattice the second condition 2. is equivalence to:
  \[ \forall a, b \in P : (z < a \wedge z < b) \Rightarrow (a \vee b < z) \]
- The meet irreducible elements are defined dually.
- We let \( J(P) \) and \( M(P) \) be the set of join-irreducible and meet-irreducible elements of \( P \).

\[ \text{Assume } z \text{ is join irreducible. We have } (z < a \wedge z < b) \Rightarrow (a \vee b < z) \] since \( a \vee b \leq z \) implies \( z = a \vee b = b \) since \( z \) is irreducible in conjunction with \( (z < a \wedge z < b) \).

Similarly, \( (z < c \wedge z < d) \Rightarrow (c \wedge d < z) \) since \( c \wedge d \leq z \) implies \( z = c \wedge d \).

(ii) Let \( a \in L \) and \( T = \{ x \in J(L) \mid x \leq a \} \). \( a \) is an upper-bound of \( T \). \( a \) is any upper bound of \( T \). We have \( a \leq c \) since otherwise \( a \not\leq c \) implies \( a \not\leq a \wedge c \). By (i), there exists \( x \in J(L) \) with \( x \leq a \) and \( a \not\leq a \wedge x \). Hence \( x \in T \) and so \( x \leq c \) since \( c \) is an upper-bound of \( T \). Thus \( x \) is a lower bound of \( \{a, c\} \) and consequently \( x \leq a \wedge c \) a contradiction. Hence \( a \leq c \) proving that \( a = \bigvee T \) in \( L \) proving that \( a = \forall a \in L : \bigvee \{ x \in J(L) \mid x \leq a \} \).

Decomposition of elements of a lattice satisfying the descending chain condition (DCC) into join irreducibles

**Theorem.** Let \( \langle L, \leq, \lor \rangle \) be a lattice satisfying the DCC.
\[ \forall a \in L : \bigvee \{ x \in J(L) \mid x \leq a \} = a \]

**Proof.** (i) \( \forall a, b \in L : (a \nleq b) \Rightarrow (\exists x \in J(L) : x \leq a \land x \nleq b) \)

Assume \( a \nleq b \). Let \( S = \{ x \in L \mid x \leq a \land x \nleq b \} \). The set \( S \) is not empty since \( a \in S \). Since \( L \) satisfies the DCC, there exists a minimal element \( z \) of \( S \). This element is join-irreducible since \( z = c \lor d \) with \( c < z \) and \( d < z \) implies, by the minimality of \( z \) that \( c \nleq S \) and \( d \nleq S \). We have \( c < z \leq a \) so \( c \leq a \) and similarly \( d \leq a \). Therefore \( c, d \nleq S \) implies \( c \nleq b \) and \( d \nleq b \). But then \( z = c \lor d \leq b \), a contradiction. Thus \( x \in J(L) \cap S \), which proves (i).

Encoding of complete join morphisms on lattices satisfying the descending chain condition (DCC) by the image of join irreducibles

**Theorem.** Let \( \langle L, \leq, \lor \rangle \) be a lattice satisfying the DCC. Let \( f \in L \rightarrow L \) be a complete join morphism. Define \( g \overset{\text{def}}{=} f \uparrow J(L) \), that is \( g \) coincide with \( f \) on join-irreducibles. Define \( f'(a) = \bigvee \{ g(x) \mid x \in J(L) \wedge x \leq a \} \)

Then \( f' = f \).

**Proof.**
\[ f'(a) = f(\bigvee \{ x \in J(L) \mid x \leq a \}) \] (\( L \) satisfies DCC)
Atoms

- Let \( \langle P, \leq, \perp \rangle \) be a poset with an infimum \( \perp \). An atom of \( P \) is \( a \in P \) such that \( \perp \prec a \) in \( P \) (i.e. \( \perp < a \) and \( \exists b \in P : \perp < b < a \)).
- The set of atoms of \( \langle P, \leq, \perp \rangle \) is denoted \( A(P) \).

Atoms and join irreducibles in Boolean lattices

**Theorem.** Let \( \langle L, \leq, \perp, \lor \rangle \) be a lattice with infimum \( \perp \). Then

1. \( \perp \prec x \in L \implies x \in \mathcal{J}(L) \)
2. If \( L \) is a boolean lattice then \( \mathcal{J}(L) \subseteq \mathcal{A}(L) \)

**Proof.**

(i) Assume \( \prec x \) and \( x = a \lor b \) with \( a \prec x \) and \( b \prec x \). Since \( \prec x \), we have \( a = b = \perp \) whence \( x = \perp \), a contradiction proving that \( x \in \mathcal{J}(L) \).

(ii) Let \( L \) be a boolean lattice and \( x \in \mathcal{J}(L) \). Assume \( \perp \leq y < x \). We have:

\[
\begin{align*}
x &= x \lor y \\
&= (x \lor y) \land (\neg y \lor y) \\
&= (x \land \neg y) \lor y
\end{align*}
\]

Since \( x \in \mathcal{J}(L) \) and \( y < x \), we must have \( x = x \land \neg y \) whence \( x \leq \neg y \). But then \( y = x \land \neg y \leq \neg y \land y = \perp \), so \( y = \perp \). This proves \( \perp \prec x \) so \( x \in \mathcal{A}(L) \) whence \( \mathcal{J}(L) \subseteq \mathcal{A}(L) \).

So in Boolean lattices it suffices to know complete join morphisms on the atoms.
Encoding of complete join morphisms on Boolean lattices satisfying the DCC by the image of atoms

- Atoms may no exist in infinite lattices (for example in \(\langle \mathbb{R}^+, \leq \rangle\)). However if they exist, they can replace join irreducible to encode complete join morphisms.
- Example:

```
\begin{center}
\begin{tikzpicture}
    \node (a) at (0,0) {\{a\}};
    \node (b) at (1,1) {\{b\}};
    \node (c) at (1,0) {\{c\}};
    \node (d) at (1,-1) {\{d\}};
    \node (e) at (2,0) {\{e\}};
    \node (f) at (2,1) {\{f\}};
    \node (g) at (2,-1) {\{g\}};
    \node (h) at (3,0) {\{h\}};
    \draw (a) -- (b);
    \draw (a) -- (c);
    \draw (a) -- (d);
    \draw (b) -- (c);
    \draw (b) -- (d);
    \draw (c) -- (d);
    \draw (e) -- (f);
    \draw (e) -- (g);
    \draw (f) -- (g);
    \draw (h) -- (f);
\end{tikzpicture}
\end{center}
```

**Theorem.** Let \(\langle L, \leq, \bot, \top \rangle\) be a Boolean lattice satisfying the DCC. Let \(f \in L \rightarrow L\) be a complete join morphism. Define \(g \overset{\text{def}}{=} f : A(L)\), that is \(g\) coincide with \(f\) on atoms. Then \(f = \lambda a. \top \{g(x) | x \in A(L) \land x \leq a\}\).

**Proof.** Immediate consequence of the previous two theorems.

---

**Closure Operators**

Kazimierz Kuratowski
Definition of an upper closure operator

- An operator on a set \( P \) is a map of \( P \) into \( P \)
- An upper closure operator \( \rho \) on a poset \( \langle P, \leq \rangle \) is
  - extensive: \( \forall x \in P : x \leq \rho(x) \)
  - monotone: \( \forall x, y \in P : (x \leq y) \Rightarrow (\rho(x) \leq \rho(y)) \)
  - idempotent: \( \rho(\rho(x)) = \rho(x) \)
- Examples:

Example of upper closure operator: reflexive transitive closure

- Let \( \Sigma \) be a set and \( t \subseteq (\Sigma \times \Sigma) \) be a relation on \( \Sigma \)
  - \( t^0 \overset{\text{def}}{=} 1_\Sigma, t^{n+1} \overset{\text{def}}{=} t \circ t^n \) : composition of relations
  - \( t^+ \overset{\text{def}}{=} \bigcup_{n \in \mathbb{N}} t^n \)
- We have
  - \( t \subseteq t^* \) : extensive
  - \( t^* \subseteq t^{t^*} \) : monotone
  - \( (t^*)^* \) : idempotent
- So that \( * \) is an upper closure operator on \( \langle \Sigma \times \Sigma, \subseteq \rangle \).
- Same for \( t^+ \)

Definition of a lower closure operator

The dual notion is that of lower closure operator, which is

- reductive: \( \forall x \in P : \rho(x) \leq x \)
- monotone
- idempotent

Topological closure operator

- A topological closure operator \( \rho \) on a poset \( \langle P, \leq, \bot, \lor \rangle \) with infimum \( \bot \) and lub \( \lor \), if any, satisfies
  - strict: \( \rho(\bot) = \bot \)
  - extensive: \( \forall x \in P : x \leq \rho(x) \)
  - join morphism: \( \forall x, y \in P : \rho(x \lor y) = (\rho(x) \lor \rho(y))^* \)
  - idempotent: \( \rho(\rho(x)) = \rho(x) \)

\( ^* \) This is the original definition given by K. Kuratowski on \( \langle \Sigma, \subseteq \rangle \) to characterize a unique topology on \( \Sigma \).

Let \( \rho \) be a topological closure operator on \( \Sigma \). Let \( T = \{ S : A \subseteq \Sigma \land \rho(A) = A \} \). Then \( T \) is a topology on \( \Sigma \) and \( \rho(A) \) is the \( T \)-closure of \( A \) for each subset \( A \) of \( \Sigma \).

\( ^* \) This implies that \( \rho \) is monotone.
Morgado Theorem (on upper closure operators)

**THEOREM.** An operator \( \rho \) on a poset \((P, \leq)\) is an upper closure operator if and only if
\[ \forall x, y \in P : x \leq \rho(y) \iff \rho(x) \leq \rho(y) \]

**PROC.** - Let \( \rho \) be an upper closure operator
\[ z \leq \rho(y) \]
\[ \implies \rho(z) \leq \rho(\rho(y)) \]  \text{(monotonicity)}
\[ \implies \rho(z) \leq \rho(y) \]  \text{(idempotence)}
\[ \implies z \leq \rho(z) \leq \rho(y) \]  \text{(extensivity)}
\[ \implies \rho(z) \leq \rho(y) \]  \text{(transitively)}

- Conversely, let \( \rho \) satisfying the above condition.

---

Fixpoints of a closure operator

The set of **fixpoints** of an operator \( f \in P \mapsto P \) on a set \( P \) is \( \{ x \mid f(x) = x \} \).

**THEOREM.** A closure operator is uniquely defined by its fixpoints.

**PROC.** Let \( \rho_1 \) and \( \rho_2 \) be two upper closure operators on a poset \((P, \leq)\) with identical fixpoints:
\[ \forall x \in P : \rho_1(x) = x \iff \rho_2(x) = x \]

We prove that \( \rho_1 = \rho_2 \).
- \( \forall x \in P : x \leq \rho_1(x) \) so \( \rho_1(x) \leq \rho_2(\rho_1(x)) \) by extensivity of \( \rho_1 \) and monotony of \( \rho_2 \).
- \( \rho_1(\rho_1(x)) = \rho_2(x) \) by idempotence so \( \rho_2(\rho_1(x)) = \rho_1(x) \) since \( \rho_1 \) and \( \rho_2 \) have the same fixpoints.
- It follows that \( \rho_2(z) \leq \rho_2(\rho_1(z)) = \rho_1(z) \)
Galois Connections

Definition of a Galois connection

- Let $\langle P, \leq \rangle$ and $\langle Q, \sqsubseteq \rangle$ be posets. A pair $\langle \alpha, \gamma \rangle$ of maps $\alpha : P \to Q$ and $\gamma : Q \to P$ is a Galois connection if and only if
  
  $\forall x \in P : \forall y \in Q : \alpha(x) \sqsubseteq y \iff x \leq \gamma(y)$

  which is written:

  $\langle P, \leq \rangle \xleftarrow{\gamma} \xrightarrow{\alpha} \langle Q, \sqsubseteq \rangle$

- $\alpha$ is the lower adjoint
- $\gamma$ is the upper adjoint

Example:
Example of Galois connection: bijection

Let $P$ and $Q$ be two sets and $b \in P \to Q$ be a one-to-one map of $p$ onto $q$ with inverse $b^{-1}$. Then

$$\langle P, \to \rangle \xleftrightarrow[b^{-1}]{b} \langle Q, \to \rangle$$

(where $\langle P, \to \rangle$ is $P$ ordered by equality)

**Proof.**

- $b(x) = y$
- $\iff x = b^{-1} \quad \text{(by def. bijection)}$

Example of Galois connection: functional abstraction

Let $C$ and $A$ be sets and $f \in C \to A$. Define

- $\alpha(X) \overset{\text{def}}{=} \{ f(x) \mid x \in X \}$
- $\gamma(Y) \overset{\text{def}}{=} \{ x \mid f(x) \in Y \}$

then

$$\langle \gamma(C), \subseteq \rangle \xleftrightarrow[\alpha]{\gamma} \langle \alpha(A), \subseteq \rangle$$

Example of Galois connections with Pre and Post

Recall that given a set $\Sigma$ and $t \subseteq \Sigma \times \Sigma$, we have defined

- $\text{post}[t] X \overset{\text{def}}{=} \{ x' \mid \exists x \in X : (x, x') \in t \}$
- $\text{pre}[t] X \overset{\text{def}}{=} \text{post}[t^{-1}] X$
- $\overline{\text{post}}[t] X \overset{\text{def}}{=} \text{post}[t] (\neg X)$
- $\overline{\text{pre}}[t] X \overset{\text{def}}{=} \text{pre}[t] (\neg X)$

- Example:
  - $C = \mathbb{Z}$, $A = \{-1, 0, +1\}$
  - $f(x) = (x < 0 \to -1, 0 \to +1)$
  - $\alpha(\{0, 1, 2\}) = \{0, +1\}$
  - $\gamma(\{0, +1\}) = \{x \in \mathbb{Z} \mid x \geq 0\} = \mathbb{N}$
We have

\[ \langle \rho(\Sigma), \subseteq \rangle \xleftarrow{\text{pre}[t]} \text{post}[t] \xrightarrow{\text{pre}[t]} \langle \rho(\Sigma), \subseteq \rangle \]

By letting \( t' = t^{-1} \), we get in the same way

\[ \langle \rho(\Sigma), \subseteq \rangle \xleftarrow{\text{post}[t]} \text{pre}[t] \xrightarrow{\text{post}[t]} \langle \rho(\Sigma), \subseteq \rangle \]

---

Example of Galois connections induced by upper closure operators

Recall Morgado's theorem for an upper closure operator on a poset \( \langle P, \leq \rangle \)

\[ \forall x, y \in P : x \leq \rho(y) \iff \rho(x) \leq \rho(y) \]

Let \( \rho(P) = \{ \rho(x) \mid x \in P \} \). This can be written as follows (with \( z = \rho(y) \))

\[ \forall x \in P : \forall z \in \rho(P) : x \leq 1_P(z) \iff \rho(x) \leq z \]

which by definition of a Galois connection implies that

\[ \langle P, \leq \rangle \xleftarrow{1_P} \langle \rho(P), \leq \rangle \]

Reciprocally, this implies that

\[ \forall x \in P : \forall z \in \rho(P) : \rho(x) \leq z \iff x \leq 1_P(z) \iff x \leq \rho(y) \text{ (z = \rho(y))} \]

so that

THEOREM. \( \rho \) is an upper closure of \( \langle P, \leq \rangle \) if and only if

\[ \langle P, \leq \rangle \xleftarrow{1_P} \langle \rho(P), \leq \rangle \]
**Unique adjoints**

**Theorem.** In a Galois connection \( \langle P, \leq \rangle \overset{\alpha}{\leftrightarrow} \langle Q, \sqsubseteq \rangle \), one adjoint uniquely determines the other, in that

\[
\alpha(x) = \bigcap \{ y \mid x \leq \gamma(y) \}, \quad \gamma(y) = \bigvee \{ x \mid \alpha(x) \sqsubseteq y \}
\]

**Proof.** - The set \( \{ y \mid \alpha(x) \sqsubseteq y \} \) has a glb which is precisely \( \alpha(x) \), so \( \alpha(x) = \bigcap \{ y \mid \alpha(x) \sqsubseteq y \} = \bigcap \{ y \mid x \leq \gamma(y) \} \) since \( \alpha(x) \sqsubseteq y \iff x \leq \gamma(y) \).

- The set \( \{ x \mid x \leq \gamma(y) \} \) has a lub which is precisely \( \gamma(y) \), so \( \gamma(y) = \bigvee \{ x \mid x \leq \gamma(y) \} = \bigvee \{ x \mid \alpha(x) \sqsubseteq y \} \) since \( \alpha(x) \sqsubseteq y \iff x \leq \gamma(y) \).

**Characteristic property of Galois connections**

- Let \( \langle P, \leq \rangle \overset{\alpha}{\leftrightarrow} \langle Q, \sqsubseteq \rangle \) then
  - \( \alpha \) is monotone
  - \( \gamma \) is monotone
  - \( 1_P \leq \gamma \circ \alpha \)
  - \( \alpha \circ \gamma \leq 1_Q \)

**Proof.** - \( \alpha(x) \sqsubseteq \gamma(y) \implies x \leq \gamma \circ \alpha(x) \)
  - \( \gamma(x) \leq \gamma(y) \implies \alpha \circ \gamma(y) \sqsubseteq y \)
  - \( x \leq y \implies x \leq \gamma \circ \alpha(x) \implies \alpha(x) \sqsubseteq \alpha(y) \)
  - \( x \sqsubseteq y \implies \alpha(\gamma(x)) \sqsubseteq y \implies \gamma(x) \leq \gamma(y) \)

**Equivalent definition of a Galois connection**

**Theorem.**

\[
\langle P, \leq \rangle \overset{\alpha}{\leftrightarrow} \langle Q, \sqsubseteq \rangle
\]

\( \iff \) \( \alpha \) is monotone \( \land \gamma \) is monotone \( \land \alpha \circ \gamma \) is reductive \( \land \gamma \circ \alpha \) is extensive

**Proof.** - We have already proved \( \iff \)

- Reciprocally, for all \( x \in P \) and \( y \in Q \)
  - \( \alpha(x) \sqsubseteq y \)

  \( \implies \gamma \circ \alpha(x) \leq \gamma(y) \) (\( \gamma \) monotone)

  \( \implies x \leq \gamma(y) \) (\( \gamma \circ \alpha \) is extensive and transitivity)

  \( \implies \alpha(x) \sqsubseteq \gamma(y) \) (\( \alpha \) is monotone)

  \( \implies \alpha(x) \sqsubseteq y \) (\( \alpha \circ \gamma \) is reductive and transitivity)
The upper adjoint of a Galois connection preserves existing lubs

**Theorem.** Let \( (P, \leq) \leftrightarrow (Q, \sqsubseteq) \) be a Galois connection and \( X \subseteq P \) such that its lub \( \bigvee X \) does exists in \( P \). Then \( \alpha(\bigvee X) \) is the lub of \( \{\alpha(x) \mid x \in X\} \) in \( Q \), that is
\[
\alpha(\bigvee X) = \bigvee \alpha(X).
\]

**Proof.**
- \( \forall x \in X : x \leq \bigvee X \) by existence of the lub \( \bigvee X \) so \( \forall x \in X : \alpha(x) \sqsubseteq \alpha(\bigvee X) \) by monotony of \( \alpha \) proving that \( \alpha(\bigvee X) \) is an upper bound of the set \( \{\alpha(x) \mid x \in X\} \in Q \).
- Let \( y \) be another upper bound of \( \{\alpha(x) \mid x \in X\} \) in \( Q \),
\[
\forall x \in X : \alpha(x) \sqsubseteq y \quad \text{(def. upper bound)} \quad \forall x \in X : x \leq \gamma(y) \quad \text{(def. Galois connection)}
\]
proving that \( \alpha(\bigvee X) \) is the least of the upper bounds of \( \{\alpha(x) \mid x \in X\} \).
- If we write \( \bigvee Y \) for the lub of \( Y \subseteq Q \) in \( (Q, \sqsubseteq) \) whenever it exists, then we have proved that \( \alpha \) preserves existing lubs, in that \( \alpha(X) = \{\alpha(x) \mid x \in X\} \).
- If \( \bigvee X \) exists in \( (P, \leq) \) then \( \bigvee \alpha(X) \) does exists in \( (Q, \sqsubseteq) \) and \( \alpha(\bigvee X) = \bigvee \alpha(X) \).

Galois connection induced by lub preserving maps

**Theorem.** Let \( \alpha : P \leftrightarrow Q \) be a complete join preserving map between posets \( (P, \leq) \) and \( (Q, \sqsubseteq) \). Define:
\[
\gamma = \lambda y \cdot \bigvee \{z \mid \alpha(z) \sqsubseteq y\}
\]
If \( \gamma \) is well-defined then
\[
(P, \leq) \leftrightarrow (Q, \sqsubseteq)
\]

\[
\square
\]
Proof. Assume that for all \( y \in Q \), \( \bigvee \{ z \mid \alpha(z) \subseteq y \} \) does exist. A counterexample is \( \alpha \) is the identity on \( P = \omega \). Then \( \omega \in \omega - 1 = Q \). \( \{ z \mid \alpha(z) \subseteq \omega \} = \omega \) but \( \bigvee \{ z \mid \alpha(z) \subseteq y \} = \bigvee \{ 0, 1, 2, \ldots \} \) does not exist in \( \omega \)!

The proof that \( \langle \alpha, \gamma \rangle \) is a Galois connection proceeds as follows:

\[
\begin{align*}
\alpha(x) & \subseteq y \\
\implies x & \in \{ z \mid \alpha(z) \subseteq y \} \\
\implies x & \leq \bigvee \{ z \mid \alpha(z) \subseteq y \} \\
\implies x & \leq \gamma(y) \\
\implies \alpha(x) & \subseteq \alpha(\bigvee \{ z \mid \alpha(z) \subseteq y \}) \quad \text{def. } \gamma \text{ and a monotone} \\
\implies \alpha(x) & \subseteq \bigvee \{ \alpha(z) \mid \alpha(z) \subseteq y \} \quad \text{def. lubs} \\
\implies \alpha(x) & \subseteq y \quad \text{def. lub}
\end{align*}
\]

Similarly, if \( \gamma \) preserves glbs and \( \alpha = \lambda x. \bigwedge \{ y \mid x \leq \gamma(y) \} \) is well-defined then \( \langle P, \leq \rangle \triangleleft_{\alpha} \langle Q, \subseteq \rangle \).

\[\text{\footnotesize More precisely, by duality, see later on page 131.}\]

Duality principle for Galois connections

**Theorem.** We have \( \langle P, \leq \rangle \triangleright_{\gamma} \langle Q, \subseteq \rangle \) iff \( \langle Q, \supseteq \rangle \triangleright_{\gamma} \langle P, \geq \rangle \) whence the dual of a Galois connection \( \langle \alpha, \gamma \rangle \) is \( \langle \gamma, \alpha \rangle \) (exchange of adjoints).

Proof.

\[
\begin{align*}
\langle P, \leq \rangle & \triangleright_{\gamma} \langle Q, \subseteq \rangle \\
\iff \forall x \in P : \forall y \in Q : \alpha(x) \subseteq y & \iff x \leq \gamma(y) \\
\iff \forall y \in Q : \forall x \in P : \gamma(y) \geq x & \iff y \subseteq \alpha(x) \\
\iff \langle Q, \supseteq \rangle & \triangleright_{\gamma} \langle P, \geq \rangle
\end{align*}
\]

Examples:

- The dual of "\( \alpha \) preserves existing lubs" is "\( \gamma \) preserves existing glbs".
- The dual of \( \alpha(x) = \bigwedge \{ y \mid x \leq \gamma(y) \} \) is \( \gamma(y) = \bigvee \{ y \mid x \subseteq \alpha(y) \} \) that is \( \gamma(y) = \bigvee \{ x \mid \alpha(x) \subseteq y \} \).
- The dual of \( \alpha \circ \gamma \circ \alpha = \alpha \) is \( \gamma \circ \alpha \circ \gamma = \gamma \)
Composition of Galois connections

**Theorem.** The composition of Galois connections is a Galois connection: if \( \langle P, \leq \rangle \xrightarrow{\gamma_1/\alpha_1} \langle Q, \sqsubseteq \rangle \) and \( \langle Q, \sqsubseteq \rangle \xrightarrow{\gamma_2/\alpha_2} \langle R, \leq \rangle \) then \( \langle P, \leq \rangle \xrightarrow{\gamma_1 \circ \gamma_2/\alpha_2 \circ \alpha_1} \langle R, \leq \rangle \)

**Proof.** Assume \( \langle P, \leq \rangle \xrightarrow{\gamma_1/\alpha_1} \langle Q, \sqsubseteq \rangle \) and \( \langle Q, \sqsubseteq \rangle \xrightarrow{\gamma_2/\alpha_2} \langle R, \leq \rangle \) then \( \forall z \in P : \forall x \in R : \alpha_2 \circ \alpha_1(z) \leq x \)

\[ \Rightarrow \alpha_1(x) \sqsubseteq \gamma_2(z) \]

\[ \Rightarrow x \leq \gamma_1 \circ \gamma_2(z) \]

The original Galois correspondences do not compose

- A Galois correspondence, as originally defined by Galois, is a pair \( (\alpha, \gamma) \) of functions on posets (originally powersets with the subset ordering, such that \( \langle P, \leq \rangle \xrightarrow{\gamma/\alpha} \langle Q, \sqsubseteq \rangle \).

---

- So \( \alpha \) is antitone: \( x \leq y \Rightarrow \alpha(x) \sqsubseteq \alpha(y) \)
- Hence when composing \( \alpha_2 \circ \alpha_1 \) is monotonic, hence not a Galois correspondence
- This justifies the introduction of Galois connections in [3] (by taking semi-dual Galois correspondences).

---

**Reference**

Galois surjections (insertions)

**Theorem.** If \((P, \leq) \xrightarrow{iota} (Q, \sqsubseteq)\) then
\[\alpha\] is onto
\[\implies \gamma\] is one-to-one
\[\implies \alpha \circ \gamma = 1_Q\]

**Proof.** Assume that \(\alpha\) is onto (\(\forall y \in Q : \exists x \in P : \alpha(x) = y\))
- Assume \(\gamma(x) = \gamma(y)\), \(\exists x', y' \in P : \alpha(x') = y\) and \(\alpha(y') = y\), and so
  \[\gamma(\alpha(x')) = \gamma(\alpha(y'))\]
\[\implies x' \leq \gamma(\alpha(y'))\]
\[\implies \alpha(x') \sqsubseteq \alpha(y')\] (by def. Galois connection)

\[\square\]

**Example of Galois surjection:**

---

\[\square\]

Galois injections

**Theorem.** By duality, if \((P, \leq) \xrightarrow{iota} (Q, \sqsubseteq)\) then
\[\gamma\] is onto
\[\implies \alpha\] is one-to-one
\[\implies \gamma \circ \alpha = 1_P\]

---

Notations:
- \((P, \leq) \xrightarrow{\gamma} (Q, \sqsubseteq)\) def \((P, \leq) \xrightarrow{\gamma} (Q, \sqsubseteq) \wedge \alpha\) is onto
- \((P, \leq) \xrightarrow{\gamma} (Q, \sqsubseteq)\) def \((P, \leq) \xrightarrow{\gamma} (Q, \sqsubseteq) \wedge \alpha\) is one-to-one
- \((P, \leq) \xrightarrow{\gamma} (Q, \sqsubseteq)\) def \((P, \leq) \xrightarrow{\gamma} (Q, \sqsubseteq) \wedge \alpha\) is bijective
Conjugate Galois connections in a Boolean algebra

**Theorem.** Let \( \langle P, \leq, 0, 1, \lor, \land, \neg \rangle \) and 
\( \langle Q, \ot, \top, \land, \lor, \neg \rangle \) be Boolean algebras and the Galois connection

\[ \langle P, \leq \rangle \underleftarrow{\gamma} \langle Q, \sqsubseteq \rangle \]

Define the conjugates \( \alpha = \neg \alpha(-x) \) and \( \gamma = \neg \gamma(-x) \).

Then

\[ \langle P, \geq \rangle \underleftarrow{\alpha} \langle Q, \sqsubseteq \rangle \]

**Proof.**

\[ \beta(x) = y \]

11 This is also called the dual, but this may cause confusion with lattice duality.

---

Example of dual Galois connections in a Boolean algebra: Pre, Post and their duals

We have

\[ \langle \wp(S), \subseteq \rangle \underleftarrow{\text{pre}[\ell]} \langle \wp(S), \subseteq \rangle \]

By conjugate/complement duality, we get

\[ \langle \wp(S), \supseteq \rangle \underleftarrow{\text{post}[\ell]} \langle \wp(S), \supseteq \rangle \]

since \( \text{pre} = \text{pre} \), hence by order duality

\[ \langle \wp(S), \subseteq \rangle \underleftarrow{\text{post}[\ell]} \langle \wp(S), \subseteq \rangle \]

---

Example of reduction of a Galois connection

- Assume a Galois connection is not a surjection, for example:

\[ \langle P, \leq \rangle \underleftarrow{\gamma} \langle Q, \sqsubseteq \rangle \]

- It is always possible to reduce \( Q \) by identifying elements with the same \( \gamma \)-image

\[ x = y \iff \gamma(x) = \gamma(y) \]
and to reduce $Q$ to the quotient $Q/\equiv$, in which case $\alpha$ becomes surjective:

$$
\alpha_{\equiv}(x) = [\alpha(x)]_{\equiv}
\gamma_{\equiv}([y]_{\equiv}) = \gamma([y])
[x]_{\equiv} = [y]_{\equiv} \iff x \equiv y \text{ on } Q/\equiv
$$

**Reduction of a Galois connection**

**Theorem.** If $\langle P, \preceq \rangle \xleftarrow{\gamma} \langle Q, \sqsubseteq \rangle$, $x \equiv y \overset{\text{def}}{=} \gamma(x) = \gamma(y)$, $\alpha(x) = [\alpha(x)]_{\equiv}$ and $\gamma([y]_{\equiv}) = \gamma(y)$, then $\langle P, \preceq \rangle \xleftarrow{\alpha_{\equiv}} \langle Q/\equiv, \sqsubseteq \rangle$

where $[x]_{\equiv} \equiv [y]_{\equiv} \overset{\text{def}}{=} x \equiv y \text{ on } Q/\equiv$

**Proof.** $\equiv$ is an equivalence relation. We let $[x]_{\equiv}$ be the equivalence class of $x \in Q$ in the quotient $Q/\equiv$.

- We have a Galois connection $\langle P, \preceq \rangle \xleftarrow{\alpha_{\equiv}} \langle Q/\equiv, \sqsubseteq \rangle$ as follows:
  $$
  \alpha(x) = [y]_{\equiv}
  \iff [\alpha(x)]_{\equiv} \subseteq [y]_{\equiv}
  \iff \alpha(x) \subseteq y
  \overset{\text{def. } \alpha_{\equiv}(x)}{=} \overset{\text{def. } \sqsubseteq_{\equiv}}{=}
  $$

**Linear Sum of Galois connections**

**Theorem.** Let $\langle Q_1, \sqsubseteq_1 \rangle \xleftarrow{\gamma_{\equiv_1}} \langle Q_1, \sqsubseteq_1 \rangle$ and $\langle Q_2, \sqsubseteq_2 \rangle \xleftarrow{\gamma_{\equiv_2}} \langle Q_2, \sqsubseteq_2 \rangle$ be Galois connections. Define the linear (ordinal) sums of posets $\langle P, \preceq \rangle \overset{\text{def}}{=} \langle Q_1, \sqsubseteq_1 \rangle \oplus \langle P_2, \preceq_2 \rangle$ and $\langle Q, \sqsubseteq \rangle \overset{\text{def}}{=} \langle Q_1, \sqsubseteq_1 \rangle \oplus \langle Q_2, \sqsubseteq_2 \rangle$ as well as $\alpha = \alpha_1 \oplus \alpha_2$ and $\gamma = \gamma_1 \oplus \gamma_2$ as follows:

$$
\alpha([0, x]) \overset{\text{def}}{=} [0, \alpha_1(x)] \quad \gamma([0, x]) \overset{\text{def}}{=} [0, \gamma_1(x)]
\alpha([1, x]) \overset{\text{def}}{=} [1, \alpha_2(x)] \quad \gamma([1, x]) \overset{\text{def}}{=} [1, \gamma_2(x)]
\gamma(0) \overset{\text{def}}{=} [0, \gamma(0)] \quad \gamma(1) \overset{\text{def}}{=} [1, \gamma(1)]
$$

then $\langle P, \preceq \rangle \xleftarrow{\alpha_{\equiv}} \langle Q, \sqsubseteq \rangle$.
Disjoint sum of Galois connections

**Theorem.** Let \( \langle P_1, \leq_1 \rangle \triangleleft \frac{\gamma_1}{\alpha_1} \langle Q_1, \sqsubseteq_1 \rangle \) and \( \langle P_2, \leq_2 \rangle \triangleleft \frac{\gamma_2}{\alpha_2} \langle Q_2, \sqsubseteq_2 \rangle \) be Galois connections. Define the disjoint sums of posets \( \langle P, \leq \rangle \overset{\text{def}}{=} \langle P_1, \leq_1 \rangle + \langle P_2, \leq_2 \rangle \) and \( \langle Q, \sqsubseteq \rangle \overset{\text{def}}{=} \langle Q_1, \sqsubseteq_1 \rangle + \langle Q_2, \sqsubseteq_2 \rangle \) as well as \( \alpha = \alpha_1 + \alpha_2 \) and \( \gamma = \gamma_1 + \gamma_2 \) as follows:

\[
\alpha((0, x)) \overset{\text{def}}{=} 0, \alpha_1(x) \overset{\text{def}}{=} \langle 0, \alpha_1(x) \rangle, \quad \alpha((1, x)) \overset{\text{def}}{=} \langle 1, \alpha_2(x) \rangle \\
\gamma((0, x)) \overset{\text{def}}{=} \langle 0, \gamma_1(x) \rangle, \quad \gamma((1, x)) \overset{\text{def}}{=} \langle 1, \gamma_2(x) \rangle
\]

then

\( \langle P, \leq \rangle \triangleleft \frac{\gamma}{\alpha} \langle Q, \sqsubseteq \rangle \)
Product of Galois connections

**Theorem.** Let \( (P_1, \leq_1) \xrightarrow{\gamma_1} (Q_1, \sqsubseteq_1) \) and \( (P_2, \leq_2) \xrightarrow{\gamma_2} (Q_2, \sqsubseteq_2) \) be Galois connections. Define the cartesian product of posets \( (P, \leq) \overset{\text{def}}{=} (P_1, \leq_1) \times (P_2, \leq_2) \) and \( (Q, \sqsubseteq) \overset{\text{def}}{=} (Q_1, \sqsubseteq_1) \times (Q_2, \sqsubseteq_2) \) as well as \( \alpha = \alpha_1 \times \alpha_2 \) and \( \gamma = \gamma_1 \times \gamma_2 \) as follows:

\[
\begin{align*}
\alpha((x, y)) & \overset{\text{def}}{=} \alpha_1(x), \alpha_2(y) \\
\gamma((x, y)) & \overset{\text{def}}{=} \gamma_1(x), \gamma_2(y)
\end{align*}
\]

then

\( (P, \leq) \xrightarrow{\frac{\gamma}{\alpha}} (Q, \sqsubseteq) \)

This can be generalized to \( (P, \leq) \xrightarrow{\frac{\gamma}{\alpha}} (Q, \sqsubseteq) \) implies

\( (P^n, \leq^n) \xrightarrow{\frac{\gamma^n}{\alpha^n}} (Q^n, \sqsubseteq^n) \) where

\[
\begin{align*}
\alpha^n((x_1, \ldots, x_n)) & = \alpha(x_1), \ldots, \alpha(x_n) \\
\gamma^n((y_1, \ldots, y_n)) & = \gamma(y_1), \ldots, \gamma(y_n)
\end{align*}
\]

Power of Galois connections

**Theorem.** Let \( (P_1, \leq_1) \xrightarrow{\gamma_1} (Q_1, \sqsubseteq_1) \) and \( (P_2, \leq_2) \xrightarrow{\gamma_2} (Q_2, \sqsubseteq_2) \) be Galois connections and \( (P_1 \xrightarrow{m} P_2, \leq_2) \) as well as \( (Q_1 \xrightarrow{m} Q_2, \sqsubseteq_2) \) be sets of monotone maps with a pointwise ordering. Then

\( (P_1 \xrightarrow{m} P_2, \leq_2) \xrightarrow{\lambda \gamma \cdot \alpha_1 \circ \alpha_2} (Q_1 \xrightarrow{m} Q_2, \sqsubseteq_2) \)

\[
\begin{align*}
\alpha & = \lambda f \cdot \alpha_2 \circ f \circ \gamma_1 \\
\gamma & = \lambda g \cdot \gamma_2 \circ g \circ \alpha_1
\end{align*}
\]
\[ \Rightarrow \; \alpha_2 \preceq f \preceq \gamma_1 \preceq_2 g \quad \text{(since } \alpha_2 \preceq \gamma_2 \text{ reductive)}
\]
\[ \Rightarrow \; \alpha(f) \preceq_2 g \quad \text{(def. } \alpha) \]
and so \( \alpha(f) \preceq_2 g \iff f \preceq_2 \gamma(g) \). \qed