Tones and Types

Michael Arntzenius, daekharel@gmail.com

18 January 2019

Abstract. Certain properties of maps between preorders (e.g. preserving equivalence) reduce to monotonicity with respect to an altered domain ordering. I dub such alterations "tones", and explore their theory. I sketch a typed λ-calculus of monotone functions, using tones to allow selective non-monotonicity.

1 Preorders

A preorder is a relation $a \leqslant b$ satisfying:

1. **Reflexivity:** $a \leqslant a$.
2. **Transitivity:** If $a \leqslant b$ and $b \leqslant c$ then $a \leqslant c$.

Preorders generalize partial orders by not requiring antisymmetry. Let $a \equiv b$ iff $a \leqslant b$ and $b \leqslant a$. Antisymmetry means $a \equiv b$ implies $a = b$. A good example preorder is "lists under containment", where $a \leqslant b$ iff every element of a is also in b. Note that $[0, 1] \equiv [1, 0]$, but $[0, 1] \neq [1, 0]$.

To a category theorist, a preorder is a "thin" category: between any two objects there is at most one morphism. I suspect much of the "tone theory" in this document, ostensibly about maps between preorders, extends to functors between categories.

2 Tones

Tones are ways a function f may respect a preorder. I will consider four tones: id, op, $\square$ (pronounced "iso"), and $\lozenge$ (pronounced "path").

Tone	*Name*	*Property of f*		
id	Monotone	$x \leqslant y$	$\Longrightarrow$	$f(x) \leqslant f(y)$
op	Antitone	$x \geqslant y$	$\Longrightarrow$	$f(x) \leqslant f(y)$
$\square$	Invariant	$x \leqslant y \wedge y \leqslant x$	$\Longrightarrow$	$f(x) \leqslant f(y)$
$\lozenge$	Bivariant	$x \leqslant y \vee y \leqslant x$	$\Longrightarrow$	$f(x) \leqslant f(y)$

Informally,

1. id is monotone (order-preserving).
2. op is antitone (order-inverting).
3. $\square$ is invariant, preserving only equivalence.
4. $\lozenge$ is bivariant: both monotone and antitone.

2.1 Tones transform orders

Fix preorders A, B. Let $\mathsf{op}\,A$ be A, ordered oppositely. Now, observe that

$$f : A \to B \text{ is antitone}$$
$$\textit{iff}$$
$$f : \mathsf{op}\,A \to B \text{ is monotone}$$

So "antitone" is a special case of "monotone"! This observation generalizes: every tone is really monotonicity with a transformation applied to the domain's ordering. So **tones transform orders**. I write TA for the preorder A transformed by the tone T, defined:

Tone	*Meaning*	*Transformation on A*		
id	same ordering	$a \leqslant b : A$	$\iff$	$a \leqslant b : \mathrm{id}\,A$
op	opposite ordering	$a \geqslant b : A$	$\iff$	$a \leqslant b : \mathsf{op}\,A$
$\Box$	induced equivalence	$a \leqslant b \wedge b \leqslant a : A$	$\iff$	$a \leqslant b : \Box A$
$\Diamond$	equivalence closure	$a \leqslant b \vee b \leqslant a : A$	$\implies$	$a \leqslant b : \Diamond A$

With this, we can state the theorem generalizing our observation:

Theorem 1 (Tones transform orders)**.**

$$f : A \to B \text{ has tone } T$$
$$\textit{iff}$$
$$f : TA \to B \text{ is monotone}$$

From this point on, when I write $f : A \to B$, I mean implicitly that f is monotone; and therefore $f : TA \to B$ means that f has tone s. Here are a few more useful properties of tones, which I invite you to verify:

Theorem 2 (Functoriality of tones)**.** If $f : A \to B$ then $f : TA \to TB$.

Theorem 3 (Tones distribute over $\times$ and $+$)**.** $T(A \times B) = TA \times TB$ and $T(A + B) = TA + TB$, where $A \times B$ and $A + B$ are the product and coproduct preorders respectively.

2.2 Understanding tone transformations

Figure 1 illustrates how each tone transforms the graph of a simple preorder: $\mathrm{id}\,A$ is identical to A; $\mathsf{op}\,A$ inverts arrows' directions; $\Box A$ isolates the strongly connected components of A; and $\Diamond A$ makes weakly connected components strong. Arrows in $\Box A$ and $\Diamond A$ are always bidirectional, as their preorders are symmetric (and thus equivalence relations).

$\Diamond A$ is a symmetric, transitive closure, the smallest preorder such that $a \leqslant b \vee b \leqslant a : A$ implies $a \leqslant b : \Diamond A$. Unlike every other tones' definition, this implication is not reversible. In figure 1, for example, $a \leqslant c : \Diamond A$, but $a \nleqslant c \wedge c \nleqslant a : A$.

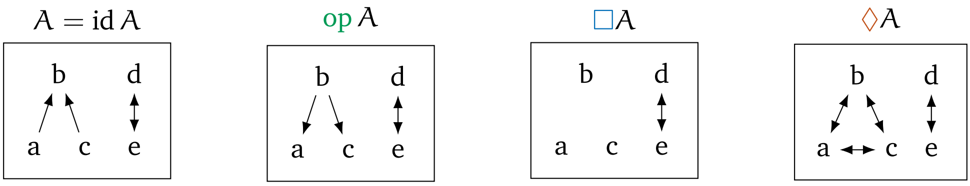
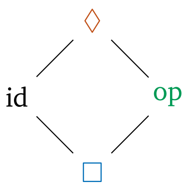


FIGURE 1. Tones applied to an example preorder



$T \wedge U$	id	$\Diamond$	op	$\Box$
id	id	id	$\Box$	$\Box$
$\Diamond$	id	$\Diamond$	op	$\Box$
op	$\Box$	op	op	$\Box$
$\Box$	$\Box$	$\Box$	$\Box$	$\Box$

UT		T: id	op	$\Box$	$\Diamond$
U	id	id	op	$\Box$	$\Diamond$
	op	op	id	$\Box$	$\Diamond$
	$\Box$	$\Box$	$\Box$	$\Box$	$\Diamond$
	$\Diamond$	$\Diamond$	$\Diamond$	$\Box$	$\Diamond$

FIGURE 2. Tone lattice, meet, and composition

2.3 Tone operators

Figure 2 defines two operators on tones:

1. Meet $T \wedge U$ is the greatest lower bound in the lattice ordered $\Box < \{\mathrm{id}, \mathrm{op}\} < \Diamond$. This finds the tone of the pairing $\langle f, g \rangle : (T \wedge U)A \to B \times C$ of two functions $f : TA \to B$ and $g : UA \to C$.
2. Composition UT gives the tone of a composed function $g \circ f : UTA \to C$ when $f : TA \to B$ and $g : UB \to C$. Equivalently, $(UT)A = U(TA)$ for any preorder A.

I sometimes write TU as $T \circ U$ for clarity. Composition binds tighter than meet, so $TU \wedge V = (T \circ U) \wedge V$. Together, $\wedge$ and $\circ$ form a semiring whose properties are given in figure 3.

3 Semantics of tones

Let's change perspective. Section 2 defines tones as function properties, then gives corresponding preorder transformations. Now, let's define tones as preorder transformations, and derive corresponding function properties.

Definition 4. $|-| : \textsc{preord} \to \textsc{set}$ is the functor taking a preorder to its set of elements.

$$
\begin{array}{lrcl}
\multicolumn{4}{c}{\textit{Properties of } \wedge} \\
\text{Associativity} & (T \wedge U) \wedge V & = & T \wedge (U \wedge V) \\
\text{Commutativity} & T \wedge U & = & U \wedge T \\
\text{Idempotence} & T \wedge T & = & T \\
\Diamond \text{ is identity} & \Diamond \wedge T & = & T \\
\Box \text{ absorbs} & \Box \wedge T & = & \Box
\end{array}
\qquad
\begin{array}{lrcl}
\multicolumn{4}{c}{\textit{Properties of } \circ} \\
\text{Associativity} & (TU)V & = & T(UV) \\
\text{Identity} & \multicolumn{3}{l}{\mathrm{id} \circ T = T = T \circ \mathrm{id}} \\
\Diamond \text{ right-absorbs} & T\Diamond & = & \Diamond \\
\Box \text{ right-absorbs} & T\Box & = & \Box \\
\mathrm{op} \text{ involutive} & \mathrm{op} \circ \mathrm{op} & = & \mathrm{id}
\end{array}
$$

$$
\begin{array}{lrcl}
\text{Left distribution} & T(U \wedge V) & = & TU \wedge TV \\
\text{Right distribution} & (T \wedge U)V & = & TV \wedge UV
\end{array}
$$

FIGURE 3. Properties of tone operators

Definition 5 (Tones)**.** A tone is a functor $T : \textsc{preord} \to \textsc{preord}$ such that $|T-| = |-|$.[1] That is, for any preorder A and monotone map f,

1. $|TA| = |A|$: tones alter a preorder's *ordering*, not its elements.
2. $|Tf| = |f|$: tones do not alter functions' behavior.

Theorem 6. Tones are closed under functor composition.

Proof. Applying definition 5, we have $|U(T-)| = |T-| = |-|$. □

3.1 The tone lattice

Preorders have a natural *subpreorder* relationship, $A \leqslant B$, given by:

$$
\begin{aligned}
A \leqslant B &\iff \leqslant_A \subseteq \leqslant_B \\
&\iff \lambda x.\, x : A \to B \\
&\iff A \subseteq B \wedge \forall (x, y)\; x \leqslant y : A \implies x \leqslant y : B
\end{aligned}
$$

This lifts pointwise to a partial order on tones:

$$
T \leqslant U \iff (\forall A)\; TA \leqslant UA
$$

Theorem 7. Preorders over a set A form a lattice.

Proof. Let $P, Q, R \subseteq A \times A$ stand for preorder relations on A. Let S^* be the transitive closure of S. Then our preorder lattice is given by:

$$
\begin{aligned}
P \wedge Q &= P \cap Q & \qquad \bot &= \{(x, x) \mid x \in A\} \\
P \vee Q &= (P \cup Q)^* & \qquad \top &= A \times A
\end{aligned}
$$

[1] A more categorical approach might require only a natural isomorphism $\iota : |T-| \simeq |-|$. I'm not yet comfortable generalizing that far.

By construction, $\bot$ is the least preorder; $\top$ the greatest; $P \wedge Q$ the greatest lower bound of P, Q; and $P \vee Q$ their least upper bound (as the least transitive relation such that $P \cup Q \subseteq P \vee Q$).[2] □

Theorem 8. Tones form a lattice.[3]

Proof. Since tones are ordered pointwise, they inherit the lattice on preorders from theorem 7, so long as the lattice operations are functorial. Restating their definitions in logical notation, given tones T, U, the tone functors $\bot, \top, T \wedge U$, and $T \vee U$ construct the smallest preorders satisfying:

$$\begin{aligned} x \leqslant y : \bot A &\iff x = y \\ x \leqslant y : \top A &\iff \top \\ x \leqslant y : (T \wedge U)A &\iff x \leqslant y : TA \wedge x \leqslant y : UA \\ x \leqslant y : (T \vee U)A &\impliedby x \leqslant y : TA \vee x \leqslant y : UA \end{aligned}$$

$\bot$ and $\top$ are functorial because *all* maps $\bot A \to \bot B$ or $\top A \to \top B$ are monotone. $T \wedge U$ is functorial by functoriality of T, U, and $\wedge$. $T \vee U$ is functorial as follows: Suppose $f : A \to B$. We wish to show $f : (T \vee U)A \to (T \vee U)B$. Suppose $x \leqslant y : (T \vee U)A$. Since $(T \vee U)A$ is a transitive closure, there exists a path $x_0, ..., x_n$ such that $x_0 = x$, $x_n = y$, and $x_i \leqslant x_{i+1} : TA \vee x_i \leqslant x_{i+1} : UA$. Fix i. Without loss of generality, let $x_i \leqslant x_{i+1} : TA$. Then $f(x_i) \leqslant f(x_{i+1}) : TA$ by functoriality of T. Thus $f(x_i) \leqslant f(x_{i+1}) : (T \vee U)A$. By transitivity $f(x) \leqslant f(y) : (T \vee U)A$. □

Conjecture 9. The syntactic definitions of $\wedge$ and $\circ$ in figure 2 agree with their semantic counterparts when applied to id, op, $\Diamond$, and $\Box$.

3.2 The TONE category

Let TONE be the category whose objects are tones and whose morphisms are natural transformations. TONE is isomorphic to the tone lattice:

Theorem 10. The following are equivalent:

$$T \leqslant U \iff \exists \eta : \textsc{tone}(T, U) \iff \exists! \eta : \textsc{tone}(T, U)$$

Proof. Expanding definitions, $T \leqslant U$ means $\lambda x.\, x : TA \to UA$ for all $A : \textsc{preord}$. By lemma 11, any $\eta : \textsc{tone}(T, U)$ is of the form $\eta_A = \lambda x.\, x : TA \to UA$. □

The crux here is that natural transformations between tones are *boring*:

[2] Interestingly, this lattice is not distributive. Let $A = \{1, 2, 3, 4\}$ and consider the preorders $P = \{1 < 2 < 4\}$, $Q = \{1 < 3\}$, and $R = \{3 < 4\}$. Then $P \wedge (Q \vee R) = \{1 < 4\}$ but $(P \wedge Q) \vee (P \wedge R)$ is discrete. TODO: is there a counterexample for the other distributive law?

[3] Since the preorder lattice is not distributive, I expect that the tone lattice isn't either, but have yet to find a counterexample.

Lemma 11. For any natural transformation $\eta : T \to U$, we have $\eta_A = \lambda x.\ x$.

Proof. Let $\mathbf{1}$ be the singleton preorder $\{\star\}$. Fix some $x : A$. Let $f : \mathbf{1} \to A = \lambda\star.\ x$. Then by naturality of η, this square commutes:

$$\begin{array}{ccc} \mathbf{1} & \xrightarrow{\eta_{\mathbf{1}}} & \mathbf{1} \\ \downarrow{\scriptstyle Tf} & & \downarrow{\scriptstyle Uf} \\ TA & \xrightarrow{\eta_A} & UA \end{array}$$

From definition 5, $Tf = f = Uf$; and since $\mathbf{1}$ is a singleton, $\eta_{\mathbf{1}} = id$, thus:

$$\begin{array}{rrcl} & \eta_A \circ Tf & = & Uf \circ \eta_{\mathbf{1}} \\ \Longrightarrow & \eta_A \circ f & = & f \\ \Longrightarrow & \eta_A(x) & = & x \end{array}$$

□

4 An aside on overline notation

An overlined and superscripted meta-expression $\overline{\Phi(i)}^{i}$ represents a sequence (of unspecified length) indexed by i. The index i clarifies which bits are repeated *with variation*, and which *without*. For example:

$$\begin{array}{lll} \overline{x_i : A_i}^{i} & \text{stands for} & x_1 : A_1,\ x_2 : A_2,\ \ldots,\ x_n : A_n \\ \overline{x_i : A}^{i} & \text{stands for} & x_1 : A,\ \ x_2 : A,\ \ \ldots,\ x_n : A \end{array}$$

This resembles the usual notation for sums of sequences, but with the bounds left implicit. For example, $\sum_i x_i y^i$ can be written $\sum \overline{x_i y^i}^{i}$ if we take $\sum$ to be a function from sequences of numbers to numbers.[4]

5 A bidirectional λ-calculus with tone inference

Figure 4 gives rules[5] for a tonal sequent calculus with a type TA representing the tone functor T applied to the type A. I adapt this into a tonal λ-calculus with bidirectional type inference. I give its syntax in figure 5 and its typing judgment forms in figure 6. TODO: explain my various abuses of notation, e.g. $T\Gamma$ and $\Gamma_1 \wedge \Gamma_2$.

---

[4] This convention is inspired by Guy Steele's talk on Computer Science Metanotation. There are videos of the talk at Clojure/conj 2017, PPoPP 2017, and Harvard University. There are also slides from Code Mesh 2017.

[5] Sent to me by Jason Reed.

$$T\overline{[U_i]\,A_i}^{\,i} = \overline{[TU_i]\,A_i}^{\,i}$$

$$\textsc{Hypothesis}\ \frac{T \leqslant \mathrm{id}}{\Gamma,\ [T]\,A \vdash A} \qquad \textsc{T-Right}\ \frac{\Gamma \vdash A}{T\Gamma \vdash TA} \qquad \textsc{T-Left}\ \frac{\Gamma,\ [TU]\,A \vdash C}{\Gamma,\ [T]\,UA \vdash C} \qquad \textsc{Weakening}\ \frac{U \leqslant T \qquad \Gamma,\ [T]\,A \vdash C}{\Gamma,\ [U]\,A \vdash C}$$

$$\textsc{Contraction}\ \frac{\Gamma,\ [T]\,A,\ [U]\,A \vdash C}{\Gamma,\ [T \wedge U]\,A \vdash C} \qquad \textsc{Cut}\ \frac{\Gamma \vdash A \qquad \Delta,\ [T]\,A \vdash C}{T\Gamma,\ \Delta \vdash C}$$

FIGURE 4. Tonal sequent calculus

variables	x		
base types	P		
tones	T, U, V	$::=$	$\mathrm{id} \mid \mathrm{op} \mid \Diamond \mid \Box$
cartesian ops	$\circledast$	$::=$	$+ \mid \times$
types	A, B, C	$::=$	$P \mid \Box A \mid \mathrm{op}\,A \mid A \to B \mid A \circledast B$
inferred terms	e	$::=$	$x \mid e\ m \mid \pi_i\ e \mid m : A$
checked terms	m, n	$::=$	$e \mid \lambda x.\ m \mid (m, n) \mid \mathrm{in}_i\ m$
			$\textbf{let}\ x = e\ \textbf{in}\ m$
			$\textbf{case}\ e\ \textbf{of}\ \overline{\mathrm{in}_i\ x \to m_i}^{\,i}$
contexts	Γ	$::=$	$\varepsilon \mid \Gamma,\ x : [T]\,A$

FIGURE 5. Syntax for the bidirectional tonal λ-calculus

TYPE CHECKING $m \Leftarrow \Gamma \vdash A$ TYPE INFERENCE $m \Rightarrow \Gamma \vdash A$ TONE ADJUNCTION $T \dashv U$ SUBTONING $T \leqslant U$

SUBTYPING $[T]\,A \leqslant B$ MODE STRIPPING $[T]\,A \prec B$

FIGURE 6. Typing judgments for the tonal λ-calculus

5.1 Typing rules

Inferred forms

$$\frac{m \Leftarrow \Gamma \vdash A}{m : A \Rightarrow \Gamma \vdash A} \qquad \frac{}{x \Rightarrow x : [\mathrm{id}]\, A \vdash A} \qquad \frac{e \Rightarrow \Gamma \vdash A \qquad [T]\, A \prec B_1 \times B_2}{\pi_i\ e \Rightarrow T\Gamma \vdash B_i}$$

$$\frac{e \Rightarrow \Gamma_1 \vdash A \qquad [T]\, A \prec B \to C \qquad m \Leftarrow \Gamma_2 \vdash B}{e\ m \Rightarrow T\Gamma_1 \wedge \Gamma_2 \vdash C}$$

Checking forms

$$\frac{e \Rightarrow \Gamma \vdash A \qquad [T]\, A \leqslant B}{e \Leftarrow T\Gamma \vdash B} \qquad \frac{m \Leftarrow \Gamma \vdash A \qquad T \in \{\square, \mathrm{op}\}}{m \Leftarrow T\Gamma \vdash TA}$$

$$\frac{e \Rightarrow \Gamma_1 \vdash A \qquad m \Leftarrow \Gamma_2, x : [T]\, A \vdash C}{\mathbf{let}\ x = e\ \mathbf{in}\ m \Leftarrow T\Gamma_1 \wedge \Gamma_2 \vdash C} \qquad \frac{m \Leftarrow \Gamma, x : [T]\, A \vdash B \qquad \mathrm{id} \leqslant T}{\lambda x.\ m \Leftarrow \Gamma \vdash A \to B}$$

$$\frac{m \Leftarrow \Gamma_1 \vdash A_1 \qquad n \Leftarrow \Gamma_2 \vdash A_2}{(m, n) \Leftarrow \Gamma_1 \wedge \Gamma_2 \vdash A_1 \times A_2} \qquad \frac{m \Leftarrow \Gamma \vdash A_i}{\mathrm{in}_i\ m \Leftarrow \Gamma \vdash A_1 + A_2}$$

$$\frac{e \Rightarrow \Gamma \vdash A \qquad [T]\, A \prec B_1 + B_2 \qquad (\forall i)\ \ m_i \Leftarrow \Gamma_i, x : [U_i]\, B_i \vdash C}{\mathbf{case}\ e\ \mathbf{of}\ \overline{\mathrm{in}_i\ x \to m_i}^{\,i} \Leftarrow \bigwedge_i (U_i T\Gamma \wedge \Gamma_i) \vdash C}$$

5.2 Tone judgments

TODO: Explain judgment $s \leqslant t$, for tone ordering, and $s \dashv t$, for tone adjunction.

$$\mathrm{id} \dashv \mathrm{id} \qquad \mathrm{op} \dashv \mathrm{op} \qquad \lozenge \dashv \square \qquad T \leqslant T \qquad \square \leqslant T \qquad T \leqslant \lozenge$$

5.3 Subtyping

TODO: Explain why we use tone-annotated subtyping.

TODO: Explain the intended algorithmic reading here. Note that we case-analyse both A and B, and argue that the order we apply the rules in shouldn't matter. Eventually I'll want to prove soundness (wrt semantics) & completeness (wrt some more declarative system).

In $[T]\, A \leqslant B$, the types A and B are inputs, and the tone T is output. In each rule

I've marked the connective being analysed in pink.

$$\textsc{refl}\quad [\mathrm{id}]\, A \leqslant A \qquad \textsc{t-right}\ \frac{[T]\, A \leqslant B}{[UT]\, A \leqslant UB} \qquad \textsc{t-left}\ \frac{[T]\, A \leqslant B \qquad U \dashv V}{[TU]\, VA \leqslant B}$$

$$\textsc{cartesian distribution}\ \frac{[T]\, A_1 \leqslant A_2 \qquad [U]\, B_1 \leqslant B_2}{[T \wedge U]\, A_1 \circledast B_1 \leqslant A_2 \circledast B_2}$$

The semantic justification for T-LEFT is as follows. Note that $\lambda x.\, x : VA \to VA$. Applying $U \dashv V$ we have $\lambda x.\, x : UVA \to A$, thus $UVA \leqslant A$, and so finally $TUVA \leqslant TA \leqslant B$. Clean up this explanation. Explain that we use adjunction rather than $st \leqslant \mathrm{id}$ directly because adjunction gives us the *most informative* result; $st \leqslant \mathrm{id}$ is declarative, $t \dashv s$ is algorithmic. Give explanations for each other rule as well.

Function subtyping, $[T]\, A_1 \to B_1 \leqslant A_2 \to B_2$, has four rules, one for each tone T produced by $[T]\, B_1 \leqslant B_2$:

$$\frac{\mathrm{id} \leqslant T \qquad [T]\, A_2 \leqslant A_1 \qquad [\mathrm{id}]\, B_1 \leqslant B_2}{[\mathrm{id}]\, A_1 \to B_1 \leqslant A_2 \to B_2}$$

$$\frac{\mathrm{op} \leqslant T \qquad [T]\, A_2 \leqslant A_1 \qquad [\mathrm{op}]\, B_1 \leqslant B_2}{[\mathrm{op}]\, A_1 \to B_1 \leqslant A_2 \to B_2}$$

$$\frac{\Diamond T = \Diamond \qquad [T]\, A_2 \leqslant A_1 \qquad [\Diamond]\, B_1 \leqslant B_2}{[\Diamond]\, A_1 \to B_1 \leqslant A_2 \to B_2} \qquad \frac{[\Diamond]\, A_2 \leqslant A_1 \qquad [\Box]\, B_1 \leqslant B_2}{[\Box]\, A_1 \to B_1 \leqslant A_2 \to B_2}$$

The premise $\Diamond T = \Diamond$ of the third rule holds for $T \neq \Box$ in our system; however, $\Diamond T = \Diamond$ captures more exactly *why* the rule is valid. TODO: Give proofs each of these rules are valid.

Are there also more precise/suggestive versions of the other premises? Can the $U \leqslant T$ constraints be turned into "composing with V is $\geqslant \mathrm{id}$", for some choice of V depending on U? Or, a hypothetical generalization of three of those rules:

$$\frac{U \dashv T \qquad U \leqslant V \qquad [V]\, A_2 \leqslant A_1 \qquad [T]\, B_1 \leqslant B_2}{[T]\, A_1 \to B_1 \leqslant A_2 \to B_2}$$

Subtyping at base types will depend on the base types you choose. Frequently, some base types' preorders will be symmetric (or even discrete, $x \leqslant y \iff x = y$), and therefore equivalence relations. Let "P equiv" hold if P's order is symmetric. Then the following refinement of REFL is useful:

$$\frac{P\ \mathrm{equiv}}{[\Diamond]\, P \leqslant P}$$

5.4 Mode stripping

$[T]\,A \prec B$ is a specialization of $[T]\,A \leqslant B$ which strips off modal operators on A, turning them into transformations on T. As in subtyping, A is an input and T an output; however, B is now an output.

$$\frac{(\forall T, B)\;\; A \neq TB}{[\mathrm{id}]\,A \prec A} \qquad\qquad \frac{[T]\,A \prec B \qquad U \dashv V}{[TU]\,VA \prec B}$$

TODO: note that we cannot strip the mode $\lozenge$. $\lozenge$ is basically a pariah; we cannot eliminate it through mode stripping, and we don't have an explicit elimination rule.

5.5 Tones and the λ rule

Here are two more general variations on the λ rule I've considered:

$$\textsc{Fn-1}\quad \frac{m \Leftarrow \Gamma, x : [T]\,A \vdash B \qquad A \leqslant [T]\,A}{\lambda x.\ m \Leftarrow \Gamma \vdash A \to B}$$

$$\textsc{Fn-2}\quad \frac{m \Leftarrow \Gamma, x : [U]\,A \vdash B \qquad [T]\,A \leqslant A \qquad \mathrm{id} \leqslant TU}{\lambda x.\ m \Leftarrow \Gamma \vdash A \to B}$$

Fn-1 requires a new judgment, $A \leqslant [T]\,B$, where A, B, T are all inputs; this doesn't seem difficult to define, but it's Yet Another Subtyping Judgment. Fn-2 avoids this, but is much less easy to explain.

However, it's not clear to me I need to generalize the λ rule. The reason I thought I did was to justify something like the following:

$$\frac{\dfrac{\vdots}{\Gamma, x : [\Box]\,A \vdash m : B}}{\Gamma \vdash \lambda x.\ m : \Box A \to B}$$

But this *could* check as follows:

$$\frac{\dfrac{\vdots}{m \Leftarrow \Gamma, x : [T]\,(\Box A) \vdash B} \qquad \mathrm{id} \leqslant T}{\lambda x.\ m \Leftarrow \Gamma \vdash \Box A \to B}$$

So the crucial question is: can we always substitute $x : [\mathrm{id}]\,\Box A$ for $x : [\Box]\,A$? It would suffice to prove the subtyping and substitution principles given in section 8.1. Can we prove these with our original, subtyping-less λ rule?

6 Pattern matching

$$\begin{array}{rcll} \text{patterns} & p, q & ::= & x \mid (p, q) \mid \mathsf{in}_i\, p \\ \text{checking terms} & m, n & ::= & \textbf{case}\ e\ \textbf{of}\ \overline{p_i \to m_i}^{\,i} \\ \text{toneless contexts} & \phi, \psi & ::= & \varepsilon \mid \phi, x : A \\ \text{judgments} & J & ::= & A \equiv B \circledast C \\ & & & p : A \vdash \phi \\ & & & p \to m \Leftarrow [T]\, A; \Gamma \vdash C \end{array}$$

The types in toneless contexts ϕ aren't annotated with tones. TODO: Explain why and when we use toneless contexts. Explain $\vec{T}\phi$ notation for a context split into its tones and its types.

6.1 Distributing modes

The pattern (x, y) matches values of type $A \times B$. But how shall we match values of type $T(A \times B)$? Well, Theorem 3 says $T(A \times B) \simeq TA \times TB$. So (x, y) can *also* match $T(A \times B)$, yielding $x : TA$ and $y : TB$. To type-check this, we'll need a judgment $A \equiv B \circledast C$ for distributing modes over a cartesian operator $\circledast$ (either $\times$ or $+$). Here A is an input and B, C are outputs.

$$\frac{}{A \circledast B \equiv A \circledast B} \qquad \frac{A \equiv B \circledast C}{TA \equiv TB \circledast TC}$$

6.2 Typing patterns

The judgment $p : A \vdash \phi$ corresponds to a PREORD-morphism $A \to 1 + \phi$. It means that the pattern p, when it matches a value of type A, produces values for ϕ's variables.

$$\frac{}{x : A \vdash x : A} \qquad \frac{A \equiv A_1 + A_2 \qquad p : A_i \vdash \phi}{\mathsf{in}_i\, p : A \vdash \phi}$$

$$\frac{A \equiv A_1 \times A_2 \qquad (\forall i)\ \ p_i : A_i \vdash \phi_i \qquad \phi_1, \phi_2 \text{ disjoint}}{(p_1, p_2) : A \vdash \phi_1, \phi_2}$$

6.3 Typing case-analysis

Typing **case** as a single rule is complicated:

$$\frac{e \Rightarrow \Gamma \vdash A \qquad (\forall i)\ \ p_i : A \vdash \phi_i \qquad (\forall i)\ \ m_i \Leftarrow \Gamma_i, \vec{T}_i \phi_i \vdash C}{\textbf{case}\ e\ \textbf{of}\ \overline{p_i \to m_i}^{\,i} \Leftarrow \bigwedge_i \left(\Gamma_i \wedge \bigwedge \vec{T}_i \Gamma\right) \vdash A}$$

We can split this up using a helper judgment, $p \to m \Leftarrow [T]\,A;\,\Gamma \vdash C$, corresponding to a morphism $TA \times \Gamma \to 1 + C$. This says that the arm $p \to m$ matches a scrutinee of type A that it uses at tone T, along with variables in Γ, to produce (if it matches) a result of type C. Then we have:

$$\frac{p : A \vdash \phi \qquad m \Leftarrow \Gamma, \vec{T}\phi \vdash C}{p \to m \Leftarrow \left[\bigwedge \vec{T}\right] A;\,\Gamma \vdash C} \qquad \frac{e \Rightarrow \Gamma \vdash A \qquad (\forall i)\ p_i \to m_i \Leftarrow [T_i]\,A;\,\Gamma_i \vdash C}{\textbf{case}\ e\ \textbf{of}\ \overline{p_i \to m_i}^{\,i} \Leftarrow \bigwedge_i (T_i\Gamma \wedge \Gamma_i) \vdash C}$$

6.4 Why do we need both stripping and distribution?

Can we also use modal distribution instead of modal stripping in our typing rules for expressions? Not quite. We can rewrite the tuple-projection rule:

$$\frac{e \Rightarrow \Gamma \vdash A \qquad A \equiv A_1 \times A_2}{\pi_i\ e \Rightarrow \Gamma \vdash A_i}$$

However, we cannot rewrite function application (shown below) this way; in general, $T(A \to B) \not\equiv TA \to TB$. (In particular for $T \in \{\square, \lozenge\}$.) So it seems there is no choice but to use subtyping.

$$\frac{e \Rightarrow \Gamma_1 \vdash A \qquad [T]\,A \prec B \to C \qquad m \Leftarrow \Gamma_2 \vdash B}{e\ m \Rightarrow T\Gamma_1 \wedge \Gamma_2 \vdash C}$$

TODO: explain why using modal stripping for pattern matching doesn't work, with the $(x, (y, z))$ versus $A \times \square(B \times C)$ example.

TODO: explain why using modal stripping rather than distribution for the tuple projection rule is fine, because of the adjunction between $\lozenge$ and $\square$.

6.5 Case analysis with guarded arms

$$\begin{array}{rcll} \text{checking expressions} & m & ::= & \textbf{case}\ e\ \textbf{of}\ \overline{p_i\ \textbf{if}\ m_i \to n_i}^{\,i} \\ \text{judgments} & J & ::= & p\ \textbf{if}\ m \to n \Leftarrow [T]\,A;\,\Gamma \vdash C \end{array}$$

$$\frac{p : A \vdash \phi \qquad m \Leftarrow \Gamma_1, \vec{T}\phi \vdash \square 2 \qquad n \Leftarrow \Gamma_2, \vec{U}\phi \vdash C}{p\ \textbf{if}\ m \to n \Leftarrow \left[\bigwedge \vec{T} \wedge \bigwedge \vec{U}\right] A;\,\Gamma \vdash C}$$

$$\frac{e \Rightarrow \Gamma \vdash A \qquad (\forall i)\ p_i\ \textbf{if}\ m_i \to n_i \Leftarrow [T_i]\,A;\,\Gamma_i \vdash C}{\textbf{case}\ e\ \textbf{of}\ \overline{p_i\ \textbf{if}\ m_i \to n_i}^{\,i} \Leftarrow \bigwedge_i (T_i\Gamma \wedge \Gamma_i) \vdash C}$$

6.6 Patterns with embedded guards

$$\begin{array}{rccl} \text{patterns} & p, q & ::= & p \ \textbf{if}\ m \\ \text{judgments} & J & ::= & p : [T]\, A \vdash \Gamma \vdash \phi \end{array}$$

Now that patterns can contain expressions, our pattern typing judgment takes an input context Γ, becoming $p : [T]\, A \vdash \Gamma \vdash \phi$. This corresponds to a morphism $\Gamma \times TA \to 1 + \phi$. However, at this point our rules get so complicated I don't trust them without a proof:

$$\frac{x \notin \phi}{x : [\mathrm{id}]\, A \vdash \vec{\Diamond}\phi \vdash x : A} \qquad \frac{A \equiv A_1 + A_2 \qquad p : [T]\, A_i \vdash \Gamma \vdash \phi}{\mathrm{in}_i\ p : [T]\, A \vdash \Gamma \vdash \phi}$$

$$\frac{A \equiv B \times C \qquad p : [T]\, B \vdash \Gamma_1 \vdash \phi \qquad q : [U]\, C \vdash \Gamma_2, \vec{V}\phi \vdash \psi}{(p, q) : \left[\left(\mathrm{id} \wedge \bigwedge \vec{V}\right) T \wedge U\right] A \vdash \mathrm{id} \wedge \bigwedge \vec{V}\Gamma_1 \wedge \Gamma_2 \vdash \phi, \psi}$$

$$\frac{p : [T]\, A \vdash \Gamma_1 \vdash \phi \qquad m \Leftarrow \Gamma_2, \vec{U}\phi \vdash \Box 2}{p\ \textbf{if}\ m : \left[T \wedge \Diamond \bigwedge \vec{U} T\right] A \vdash \mathrm{id} \wedge \Diamond \bigwedge \vec{U}\Gamma_1 \wedge \Diamond\Gamma_2 \vdash \phi}$$

Now we update the rules for $p \to m \Leftarrow [T]\, A; \Gamma \vdash C$ to pass through Γ to the pattern:

$$\frac{p : [T]\, A \vdash \Gamma_1 \vdash \phi \qquad m \Leftarrow \Gamma_2, \vec{U}\phi \vdash C}{p \to m \Leftarrow [T]\, A; \bigwedge \vec{U}\Gamma_1 \wedge \Gamma_2 \vdash C}$$

7 Declarative rules

This is where I'm stashing important inference rules, stated in a way that makes them obviously valid, but leaves non-obvious how to algorithmically check them.

7.1 Subtyping and type equivalence

Type equivalence $A \equiv B$ is a synonym for $A \leqslant B \wedge B \leqslant A$. Let $A \leqslant B$ be the preorder generated by:

$$\frac{T \leqslant U}{TA \leqslant UA} \qquad \frac{A \leqslant B}{TA \leqslant TB} \qquad T(UA) \equiv (TU)A \qquad T(A \circledast B) \equiv TA \circledast TB$$

$$\mathrm{op}\,(A \to B) \equiv \mathrm{op}\, A \to \mathrm{op}\, B \qquad A \to \Box B \equiv \Box(A \to \Box B)$$

$$\Box(A \to B) \leqslant \Box A \to \Box B \qquad \frac{A_1 \leqslant B_1 \qquad A_2 \leqslant B_2}{A_1 \circledast A_2 \leqslant B_1 \circledast B_2} \qquad \frac{A_2 \leqslant A_1 \qquad B_1 \leqslant B_2}{A_1 \to B_1 \leqslant A_2 \to B_2}$$

Conjecture 12. This judgment is complete for subpreordering relationships of the form $\mathsf{T}A \leqslant B$ where $\lozenge$ does not occur in A or B (but may occur in T).

TODO: check we can derive the algorithmic function subtyping rules. $\square(A \to B) \leqslant \square A \to \square B$ handles one of the cases; $\mathsf{op}\,(A \to B) \equiv \mathsf{op}\,A \to \mathsf{op}\,B$ handles another; what about the last one?

As I originally conceived of this system, there was no type $\lozenge A$ internalizing the $\lozenge$ tone, so I imagine it's not complete for types of that form. In fact, there are *no* rules above about $\lozenge$ specifically.

Here are some more valid rules, but do I need them?

$$\mathrm{id}\,A \equiv A \qquad \frac{\mathsf{T}, \mathsf{U} \in \{\square, \lozenge\}}{A \to \mathsf{U}B \equiv \mathsf{T}(A \to \mathsf{U}B)} \qquad \frac{\mathsf{T} \in \{\square, \lozenge\} \qquad \mathsf{U} \dashv \mathsf{V}}{\mathsf{T}(\mathsf{U}A \to B) \equiv \mathsf{T}(A \to \mathsf{V}B)}$$

8 Metatheory

8.1 Weakening, subtyping, and substitution

We wish to prove admissible the following rules:

$$\textsc{tone weakening}\ \frac{m \Leftarrow \Gamma \vdash A}{m \Leftarrow \Gamma \wedge \Gamma' \vdash A} \qquad \textsc{subtyping left}\ \frac{m \Leftarrow \Gamma, x : [\mathsf{T}]\, A \vdash C \qquad \mathsf{T}A \leqslant \mathsf{U}B}{m \Leftarrow \Gamma, x : [\mathsf{U}]\, B \vdash C}$$

$$\textsc{subtyping right}\ \frac{m \Leftarrow \Gamma \vdash A \qquad \mathsf{T}A \leqslant B}{m \Leftarrow \mathsf{T}\Gamma \vdash B} \qquad \textsc{substitution}\ \frac{e \Rightarrow \Gamma_1 \vdash A \qquad \mathsf{T}A \leqslant \mathsf{U}B \qquad m \Leftarrow \Gamma_2, x : [\mathsf{U}]\, B \vdash C}{m[e/x] \Leftarrow \mathsf{T}\Gamma_1 \wedge \Gamma_2 \vdash C}$$

TODO: Doesn't TONE WEAKENING follow from SUBTYPING LEFT?