Introduction to Program Semantics — Dexter Kozen

Lecture 4 - Axiomatic and Probabilistic Semantics *June 27, 2026*

1 States, Tests, and Observation

Axiomatic semantics asks: *what can be observed about a program state?* We fix a set of states S. For the language IMP, a state is an environment

$$s \in \mathsf{Env} = \mathsf{Var} \to \mathbb{N}.$$

The things we can "observe" are syntactically described properties called *tests*. For IMP these are the boolean expressions:

$$b \ ::= \ a_1 \leq a_2 \mid a_1 = a_2 \mid b_1 \wedge b_2 \mid \neg b \mid \cdots$$

A semantics fixes a *satisfaction relation* $s \vDash B$ between states and tests. For IMP,

$$s \vDash a_1 \leq a_2 \iff \langle a_1 \leq a_2,\, s\rangle \downarrow_b \mathsf{true}.$$

Why this matters. Axiomatic semantics only talks about what is observable *before* and *after* a program runs, by means of tests.

2 Hoare Logic

2.1 Hoare triples

A *Hoare triple* is a judgement of the form

$$\{B\}\, p\, \{C\}$$

read informally as:

> "If the precondition B holds immediately before executing p, and if p terminates, then the postcondition C holds immediately after."

Compiled By:
Yanjun Chen, Kunha Kim, Jonathon Yallop

2.2 Inference rules

$$\frac{}{\{B[a/x]\}\, x := a\, \{B\}} \qquad \frac{}{\{B\}\, \mathsf{skip}\, \{B\}}$$

$$\frac{\{B\}\, p\, \{C\} \qquad \{C\}\, q\, \{D\}}{\{B\}\, p\,;\, q\, \{D\}} \qquad \frac{\{B \wedge C\}\, p\, \{D\} \qquad \{\neg B \wedge C\}\, q\, \{D\}}{\{C\}\, \mathsf{if}\ B\ \mathsf{then}\ p\ \mathsf{else}\ q\, \{D\}}$$

$$\frac{\{B \wedge C\}\, p\, \{C\}}{\{C\}\, \mathsf{while}\ B\ \mathsf{do}\ p\, \{\neg B \wedge C\}}$$

Remark. The assignment rule is "read backwards": to guarantee B after $x := a$, require B with a substituted for x before. This is the first hint that Hoare logic is fundamentally a *backward* reasoning principle. In the while rule, C is the *loop invariant*; it must hold before the loop, be preserved by the body when the guard is true, and on exit it is $\neg B$.

2.3 Relative Completeness

Theorem 1 (Cook, 1974)**.** *Hoare logic is* relatively complete.

"Relatively complete" means: assuming an oracle that decides all true assertions of the underlying assertion language about the domain of computation expressible, every valid partial-correctness triple is derivable. The crucial requirement on the assertion language is: it can express the *weakest liberal precondition* of any program with respect to any postcondition.

2.4 Weakest (Liberal) Preconditions

Definition 1. The *weakest liberal precondition* $\mathrm{wlp}(p, B)$ is the set of states s such that *if* p run from s halts, *then* the resulting state satisfies B. The *weakest precondition* additionally requires termination:

$$\mathrm{wp}(p, B) \;\Leftrightarrow\; \mathrm{wlp}(p, B) \wedge (p \text{ halts}).$$

The defining equations (read as definitions *by recursion on* p):

$$\begin{aligned}
\mathrm{wlp}(\mathsf{skip}, B) &= B \\
\mathrm{wlp}(x := a, B) &= B[a/x] \\
\mathrm{wlp}(p\,;\, q, B) &= \mathrm{wlp}(p, \mathrm{wlp}(q, B)) \\
\mathrm{wlp}(\mathsf{if}\ B\ \mathsf{then}\ p\ \mathsf{else}\ q, C) &= \big(B \Rightarrow \mathrm{wlp}(p, C)\big) \wedge \big(\neg B \Rightarrow \mathrm{wlp}(q, C)\big)
\end{aligned}$$

For loops we use induction on iteration. If an assertion I satisfies

$$B \wedge I \Rightarrow \mathrm{wlp}(p, I) \qquad \text{and} \qquad \neg B \wedge I \Rightarrow C,$$

then

$$I \Rightarrow \mathrm{wlp}(\mathsf{while}\ B\ \mathsf{do}\ p,\ C)$$

Remark. Compare the rule for $p\,;\,q$ with the Hoare rule for sequencing: in both cases, the postcondition is "pulled back" through the program from the end to the beginning. The flow of information is opposite to the flow of control.

3 Dynamic Logic

Dynamic logic embeds Hoare-style reasoning into a modal logic in which *programs are modalities*. Given the denotation $[\![p]\!]$ of a program as a binary relation on states:

$$\begin{aligned} s \vDash [p]A &\iff \forall t.\ (s,t) \in [\![p]\!] \Rightarrow t \vDash A, \\ s \vDash \langle p\rangle A &\iff \exists t.\ (s,t) \in [\![p]\!] \wedge t \vDash A. \end{aligned}$$

So $[p]A$ says "every terminating run of p from s ends in A" (a partial-correctness statement), and $\langle p\rangle A$ says "*some* run reaches A".

Axioms

In addition to propositional logic:

$$\begin{aligned} [p]A &\iff \neg\langle p\rangle\neg A \\ [p](A \Rightarrow B) &\Rightarrow [p]A \Rightarrow [p]B \\ [A]B &\iff A \Rightarrow B && \text{(test programs)} \\ [p+q]A &\iff [p]A \wedge [q]A \\ [p\,;\,q]A &\iff [p][q]A \\ [p^*]A &\iff A \wedge [p][p^*]A \\ \big(A \wedge [p^*](A \Rightarrow [p]A)\big) &\Rightarrow [p^*]A \end{aligned}$$

Rules of inference (modus ponens and necessitation):

$$\frac{A \qquad A \Rightarrow B}{B} \qquad \frac{A}{[p]A}$$

3.1 Predicate Transformers

The operators $[p]$, $\langle p\rangle$, $\mathrm{wlp}(p,-)$, and $\wp(p,-)$ are all *predicate transformers*: they map postconditions to preconditions. Two key bridges between Hoare logic, dynamic logic, and the wlp formula-

tion:

$$A \Rightarrow [p]B \iff \{A\}\, p\, \{B\}$$
$$[p]B = \mathrm{wlp}(p, B)$$

4 Duality: Forward Execution vs. Backward Prediction

$$s \vDash [p]A \iff \mathcal{C}[\![p]\!]s \vDash A$$

- On the **left**, we have moved on the *predicate* side: the postcondition A has been transformed *backward* by the operator $[p]$ into a new predicate $[p]A$, and we check whether the original state s satisfies it.
- On the **right**, we have moved on the *state* side: the state s has been transformed *forward* by $\mathcal{C}[\![p]\!]$ into $\mathcal{C}[\![p]\!]s$, and we check whether the resulting state(s) satisfy the original postcondition A.

The equivalence says these two computations give the same truth value. That is the duality: $\mathcal{C}[\![p]\!]$ and $[p]$ are *transposes* of each other.

5 Probabilistic Semantics

So far we have been using deterministic reasoning, which tends to be more truth oriented, focusing on yes or no questions. Especially on things such as correctness, termination, and worst-case complexity (whether a program always has a specific runtime).

For probablistic reasoning, we will be more concerned with quantitative assertions. We want to ask questions such as the following:

- *Correctness:* What is the probability that the postcondition is satisfied?
- *Termination:* What is the likelihood that this program halts?
- *Average-case complexity:* What is the expected halting time, given this input distribution on inputs of length n?

5.1 Adding Probability

To add probabilistic reasoning, we take each of the following objects from the deterministic logic and replace them with the following:

states	$\Rightarrow$	measures
predicates	$\Rightarrow$	measurable functions
$\models$	$\Rightarrow$	$\int$
binary relations	$\Rightarrow$	Markov kernels
predicate transformers	$\Rightarrow$	measurable function transformers

In practice, when we add these to IMP, we will also need to add random variables. For this we will need a *random assignment* $x := \textbf{rnd}$ where each call assigns a random value to x. Samples are chosen according to some fixed probability distribution and are independent.

Examples:

- $x := \textbf{rnd}$ assigns 0 or 1 to x, each with probability $\frac{1}{2}$ (fair coin)
- $x := \textbf{rnd}$ assigns a random real in $[0, 1]$ to x according to the uniform distribution on $[0, 1]$ — the sample will be in $[a, b]$ with probability $b - a$ (note that this means any individual real number has probability 0).

5.2 Measurable Spaces and Functions

The state space must be a *measurable space* $(S, \mathcal{B})$, where S is a set and $\mathcal{B} \subseteq 2^S$ are the *measurable sets* or *events*. Elements of $\mathcal{B}$ are the observable events — all tests must denote a set in $\mathcal{B}$. $\mathcal{B}$ must form a *σ-algebra* on S — a Boolean algebra closed under countable union. A function $f : (S_1, \mathcal{B}_1) \to (S_2, \mathcal{B}_2)$ is *measurable* if the inverse image of any measurable set in $\mathcal{B}_2$ is measurable in $\mathcal{B}_1$.

Examples:

- Discrete space $(A, 2^A)$, A any finite or countable set.
- $(\mathbb{R}, \mathcal{B})$, where $\mathcal{B}$ are the Borel sets, the smallest σ-algebra on $\mathbb{R}$ containing all intervals.
- $([0, 1], \{A \cap [0, 1] \mid A \in \mathcal{B}\})$.
- $(\mathbb{R}^n, \mathcal{B}^{(n)}) = (\mathbb{R}, \mathcal{B}) \times \cdots \times (\mathbb{R}, \mathcal{B})$ — $\mathcal{B}^{(n)}$ is the smallest σ-algebra containing all measurable rectangles $A_1 \times \cdots \times A_n$, $A_i \in \mathcal{B}$.
- Cantor space $\{0, 1\}^\omega$, with Borel sets generated by the intervals $\{\alpha \in \{0, 1\}^\omega \mid x \leq \alpha\}$ for $x \in \{0, 1\}^*$.

Remark. We cannot take all sets 2^S as the measurable sets because for $[0, 1]$, it breaks ZFC.

5.3 Measures

A *measure* on $(S, \mathcal{B})$ is a map $\mu : \mathcal{B} \to \mathbb{R}$ such that for any countable collection $\mathcal{A}$ of pairwise disjoint measurable sets,

$$\mu\Big(\bigcup \mathcal{A}\Big) = \sum_{A \in \mathcal{A}} \mu(A), \qquad \mu(\emptyset) = 0.$$

- A measure is a *probability measure* if all $\mu(A) \geq 0$ and $\mu(S) = 1$.
- A measure is a *subprobability measure* if all $\mu(A) \geq 0$ and $\mu(S) \leq 1$.
- The measures on $(S, \mathcal{B})$ form a *Banach space* $\mathcal{M}$ over $\mathbb{R}$ with pointwise operations $(a\mu + b\nu)(A) = a\mu(A) + b\nu(A)$ and total variation norm $\|\mu\| = \sup_{A \in \mathcal{B}} |\mu(A)|$.

Examples

- *Dirac (point mass) measure* on a point s:

$$\delta_s(A) \triangleq \begin{cases} 1, & s \in A, \\ 0, & s \notin A. \end{cases}$$

- *Uniform (Lebesgue) measure* λ on $[0, 1]$ generated by $\lambda([a, b)) = b - a$.
- *Uniform (Lebesgue) measure* λ on Cantor space $\{0, 1\}^\omega$ generated by $\lambda(\{\alpha \mid x \leq \alpha\}) = 2^{-|x|}$.

5.4 Markov Kernels

- A *Markov kernel* is a function $P : S \times \mathcal{B} \to [0, 1]$ such that:
  - for fixed $s \in S$, $P(s, -)$ is a subprobability measure $\mathcal{B} \to [0, 1]$;
  - for fixed $B \in \mathcal{B}$, $P(-, B)$ is a measurable function $S \to [0, 1]$.
- Probabilistic analog of binary relations.
- A program p will denote a Markov kernel $[\![p]\!] : S \times \mathcal{B} \to [0, 1]$.
- $[\![p]\!](s, B)$ is the probability of event B on output, starting in input state s.

5.5 Composition of Markov Kernels

To compose two Markov kernels we integrate "up the middle" using Lebesgue integration:

$$(P\,;\,Q)(s, A) \triangleq \int_t P(s,\, dt) \cdot Q(t, A)$$

- Probabilistic analog of relational composition or Boolean matrix multiplication.
- Just multiplication of stochastic matrices in the discrete case.

5.6 Lifting

A Markov kernel can be *lifted* to a linear map $\mathcal{M} \to \mathcal{M}$ via integration:

$$P(\mu)(A) \triangleq \int_t \mu(dt) \cdot P(t, A)$$

- This map is linear and continuous on $\mathcal{M}$.
- This is Kleisli lifting again.

5.7 Probabilistic Conditional

For **if** B **then** p **else** q with input measure μ:

Split μ into the part supported on B and the part on $\overline{B}$:

$$\mu_B = \text{ measure restricted to } B, \qquad \mu_{\overline{B}} = \text{ measure restricted to } \overline{B}.$$

Run p on μ_B and q on $\mu_{\overline{B}}$, then recombine:

$$[\![\textbf{if } B \textbf{ then } p \textbf{ else } q]\!](\mu) = p(\mu_B) + q(\mu_{\overline{B}})$$

6 Measure Transformer Semantics

$$\begin{aligned}
[\![x := t]\!](s) &\triangleq \delta_{s[s(t)/x]} && \text{(point mass on } s[s(t)/x]) \\
[\![p\,;\, q]\!](s) &\triangleq [\![q]\!]([\![p]\!](s)) \\
[\![\textbf{if } B \textbf{ then } p \textbf{ else } q]\!](s) &\triangleq \begin{cases} [\![p]\!](s), & s \in B, \\ [\![q]\!](s), & s \notin B. \end{cases}
\end{aligned}$$

Define $[\![B]\!](s, A) \triangleq \begin{cases} 1, & s \in A \cap B, \\ 0, & \text{otherwise.} \end{cases}$

Curried and lifted: $[\![B]\!](\mu) = \mu_B$.

$$[\![\textbf{if } B \textbf{ then } p \textbf{ else } q]\!] = [\![B\,;\, p]\!] + [\![\overline{B}\,;\, q]\!]$$

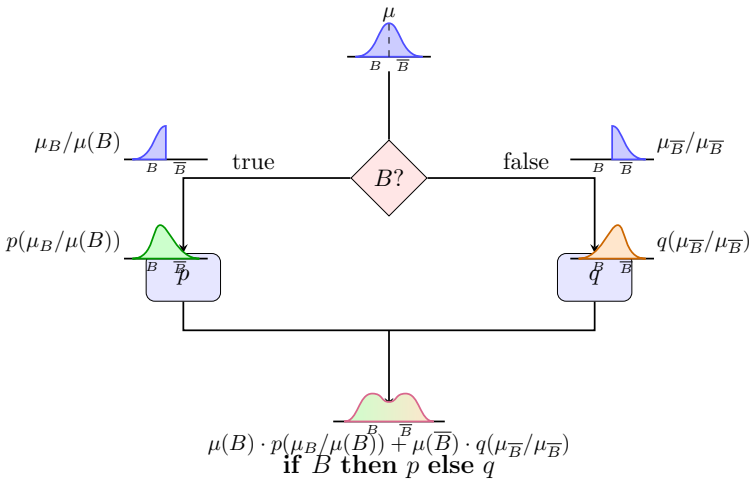

Figure 1: Semantic Flow of Transforming and Recombining Subprobability Measures

6.1 Measure Transformer Semantics

$$
\begin{aligned}
[\![x := t]\!](s) &\triangleq \delta_{s[s(t)/x]} && \text{(point mass on } s[s(t)/x]\text{)} \\
[\![p\,;\,q]\!](s) &\triangleq [\![q]\!]([\![p]\!](s)) \\
[\![\textbf{if } B \textbf{ then } p \textbf{ else } q]\!](s) &\triangleq \begin{cases} [\![p]\!](s), & s \in B, \\ [\![q]\!](s), & s \notin B. \end{cases}
\end{aligned}
$$

Define $[\![B]\!](s, A) \triangleq \begin{cases} 1, & s \in A \cap B, \\ 0, & \text{otherwise.} \end{cases}$

Curried and lifted: $[\![B]\!](\mu) = \mu_B$.

$$[\![\textbf{if } B \textbf{ then } p \textbf{ else } q]\!] \;=\; [\![B\,;\,p]\!] + [\![\overline{B}\,;\,q]\!]$$

6.2 Measure Transformer Semantics: Random Assignment

$$[\![x_i := \textbf{rnd}]\!](s_1, \ldots, s_n,\ A_1 \times \cdots \times A_n) \;\triangleq\; \left(\prod_{j \neq i} \delta_{s_j}(A_j)\right) \cdot \lambda(A_i)$$

where λ is the uniform measure on $[0, 1)$.

6.3 Measure Transformer Semantics: While Loop

$$\begin{aligned}[\![\textbf{while } B \textbf{ do } p]\!] &\triangleq [\![\overline{B}]\!] + [\![B\,;\,p\,;\,\overline{B}]\!] + [\![B\,;\,p\,;\,B\,;\,p\,;\,\overline{B}]\!] + \cdots \\ &= \sum_n [\![(B\,;\,p)^n\,;\,\overline{B}]\!] \\ &= [\![(B\,;\,p)^{*}\,;\overline{B}]\!]\end{aligned}$$

where $[\![p^*]\!] \triangleq \sum_n [\![p^n]\!]$ (though $*$ can produce unbounded measures).

6.4 Predicate Transformer Semantics

- Previously: whether a state satisfies a predicate, $s \models A$.
- With probability distributions μ, we can ask about the probability of an event A — this is $\mu(A)$.
- More generally, we can ask about the *expected value* of a measurable function $f : \mathbb{R}^n \to \mathbb{R}$:

$$\int_t \mu(dt) \cdot f(t)$$

- This is the probabilistic analog of $s \models A$.

7 Predicate Transformer Semantics

The function

$$\langle p \rangle f \triangleq \int_t [\![p]\!](-, dt) \cdot f(t)$$

gives the expected value of f after executing p.

- This is a generalized predicate transformer $\langle p \rangle : F \to F$, where F is the space of bounded measurable functions.
- Currying and lifting:

$$\mathbb{R}^n \times \mathcal{B}^{(n)} \to [0,1] \xrightarrow{\text{curry}} \mathcal{B}^{(n)} \to F \xrightarrow{\text{lift}} F \to F$$

- F is a Banach space over $\mathbb{R}$ with pointwise operations and norm $\|f\| = \sup_{\|s\|=1} |f(s)|$.

7.1 Predicate Transformer Semantics: PPDL

A logic based on this idea — PPDL:

- $\langle p\rangle A$ = probability that A is satisfied on output — probabilistic analog of weakest precondition.
- $\langle p\rangle f$ = expected value of random variable f on output.
- $[p]A \triangleq 1 - \langle p\rangle(1-A) = [p]0 + \langle p\rangle A$
- $[p]A$ = probability that A is satisfied on output *or* p doesn't halt — probabilistic analog of weakest liberal precondition.
- $A \le [p]B$ = probabilistic analog of Hoare triple $\{A\}\,p\,\{B\}$.

7.2 Axioms of PPDL

$$\begin{aligned}
\langle ap + bq\rangle f &= a\langle p\rangle f + b\langle q\rangle f \\
\langle p\rangle(af + bg) &= a\langle p\rangle f + b\langle p\rangle g \\
0 \le f &\Rightarrow 0 \le \langle p\rangle f \\
f \le g &\Rightarrow \langle p\rangle f \le \langle p\rangle g \\
\langle p\,;\,q\rangle f &= \langle p\rangle\langle q\rangle f \\
\langle B\rangle f &= Bf
\end{aligned}
\qquad\qquad
\begin{aligned}
[p]f &= [p]0 + \langle p\rangle f \\
0 \le f &\Rightarrow \langle p^*\rangle f = f + \langle p\rangle\langle p^*\rangle f \\
0 \le f &\Rightarrow \langle p^*\rangle f = f + \langle p^*\rangle\langle p\rangle f \\
0 \le f + \langle p\rangle g \le g &\Rightarrow \langle p^*\rangle f \le g \\
f \le 1 &\Rightarrow \langle p\rangle f \le 1 \\
0 \le B \le BB &\le 1
\end{aligned}$$

7.3 Duality

Write:

$$\begin{aligned}
(\mu,\ f)\ \text{ for }\ & \int_t \mu(dt)\cdot f(t) && \text{(Lebesgue integral)} \\
\mu\langle p\rangle\ \text{ for }\ & \int_t \mu(dt)\cdot [\![p]\!](t,-) && \text{(measure transformer)} \\
\langle p\rangle f\ \text{ for }\ & \int_t [\![p]\!](-,dt)\cdot f(t) && \text{(predicate transformer)}
\end{aligned}$$

Then:

$$(\mu,\ \langle p\rangle f)\ =\ (\mu\langle p\rangle,\ f)$$

7.4 Denotational Semantics

An ω-*CPO* is a poset $(A, \sqsubseteq)$ such that every countable chain has a supremum.

Theorem 2 (Saheb-Djahromi 1980)**.** *The measures on the Borel sets of the Scott topology on an ω-CPO form an ω-CPO.*

7.5 Measure Transformer Semantics: Random Assignment

$$[\![x_i := \mathbf{rnd}]\!](s_1, \ldots, s_n,\ A_1 \times \cdots \times A_n) \triangleq \left(\prod_{j \neq i} \delta_{s_j}(A_j)\right) \cdot \lambda(A_i)$$

where λ is the uniform measure on $[0, 1)$.

7.6 Measure Transformer Semantics: While Loop

$$\begin{aligned}
[\![\mathbf{while}\ B\ \mathbf{do}\ p]\!] &\triangleq [\![\overline{B}]\!] + [\![B\,;\,p\,;\,\overline{B}]\!] + [\![B\,;\,p\,;\,B\,;\,p\,;\,\overline{B}]\!] + \cdots \\
&= \sum_n [\![(B\,;\,p)^n\,;\,\overline{B}]\!] \\
&= [\![(B\,;\,p)^{*\,;\overline{B}}]\!]
\end{aligned}$$

where $[\![p^*]\!] \triangleq \sum_n [\![p^n]\!]$ (though $*$ can produce unbounded measures).

7.7 Predicate Transformer Semantics

What can you observe about a state?

- Previously: whether a state satisfies a predicate, $s \models A$.
- With probability distributions μ, we can ask about the probability of an event A — this is $\mu(A)$.
- More generally, we can ask about the *expected value* of a measurable function $f : \mathbb{R}^n \to \mathbb{R}$:

$$\int_t \mu(dt) \cdot f(t)$$

- This is the probabilistic analog of $s \models A$.

8 Predicate Transformer Semantics

The function

$$\langle p\rangle f \triangleq \int_t [\![p]\!](-, dt)\cdot f(t)$$

gives the expected value of f after executing p.

- This is a generalized predicate transformer $\langle p\rangle : F \to F$, where F is the space of bounded measurable functions.
- Currying and lifting:

$$\mathbb{R}^n \times \mathcal{B}^{(n)} \to [0,1] \xrightarrow{\text{curry}} \mathcal{B}^{(n)} \to F \xrightarrow{\text{lift}} F \to F$$

- F is a Banach space over $\mathbb{R}$ with pointwise operations and norm $\|f\| = \sup_{\|s\|=1} |f(s)|$.

8.1 Predicate Transformer Semantics: PPDL

A logic based on this idea — PPDL:

- $\langle p\rangle A$ = probability that A is satisfied on output — probabilistic analog of weakest precondition.
- $\langle p\rangle f$ = expected value of random variable f on output.
- $[p]A \triangleq 1 - \langle p\rangle(1-A) = [p]0 + \langle p\rangle A$
- $[p]A$ = probability that A is satisfied on output *or* p doesn't halt — probabilistic analog of weakest liberal precondition.
- $A \le [p]B$ = probabilistic analog of Hoare triple $\{A\}\, p\, \{B\}$.

8.2 Axioms of PPDL

$$\langle ap + bq\rangle f = a\langle p\rangle f + b\langle q\rangle f \qquad [p]f = [p]0 + \langle p\rangle f$$

$$\langle p\rangle(af + bg) = a\langle p\rangle f + b\langle p\rangle g \qquad 0 \le f \Rightarrow \langle p^{*\rangle} f = f + \langle p\rangle\langle p^{*\rangle} f$$

$$0 \le f \Rightarrow 0 \le \langle p\rangle f \qquad 0 \le f \Rightarrow \langle p^{*\rangle} f = f + \langle p^{*\rangle}\langle p\rangle f$$

$$f \le g \Rightarrow \langle p\rangle f \le \langle p\rangle g \qquad 0 \le f + \langle p\rangle g \le g \Rightarrow \langle p^{*\rangle} f \le g$$

$$\langle p\,;\, q\rangle f = \langle p\rangle\langle q\rangle f \qquad f \le 1 \Rightarrow \langle p\rangle f \le 1$$

$$\langle B\rangle f = Bf \qquad 0 \le B \le BB \le 1$$

8.3 Duality

Write:

$$\begin{aligned}
(\mu,\ f) \quad &\text{for} \quad \int_t \mu(dt) \cdot f(t) && \text{(Lebesgue integral)} \\
\mu\langle p\rangle \quad &\text{for} \quad \int_t \mu(dt) \cdot [\![p]\!](t,-) && \text{(measure transformer)} \\
\langle p\rangle f \quad &\text{for} \quad \int_t [\![p]\!](-,dt) \cdot f(t) && \text{(predicate transformer)}
\end{aligned}$$

Then:

$$\big(\mu,\ \langle p\rangle f\big) \ = \ \big(\mu\langle p\rangle,\ f\big)$$

8.4 Denotational Semantics

An ω-*CPO* is a poset $(A, \sqsubseteq)$ such that every countable chain has a supremum.

Theorem 3 (Saheb-Djahromi 1980)**.** *The measures on the Borel sets of the Scott topology on an ω-CPO form an ω-CPO.*