Formalisms Every Computer Scientist Should Know

Contents

Class 1

0.1 Tools

- Lean or Coq
- CVC5 or Z3
- possibly a model checker
- professional version of ChatGPT

0.2 Syllabus

1. MATH (“Informalism”)
    - proofs (natural deduction)
    - fixpoints (induction, coinduction)
2. DECLARATIVE LOGIC
    - syntax (rules) vs. semantics (models)
    - propositional, predicate, modal logic
    - decision procedures (SAT, SMT)
3. FUNCTIONS
    - λ calculus, typed λ
    - SOS (structured operational semantics), rewriting
    - “propositions-as-types” (connection to logic)
4. PROCESSES (CONCURRENT)*
    - CCS, Petri nets
    - (bi-) simulation
5. CIRCUITS*
    - boolean, sequential, dataflow (Kahn nets)
    - interfaces

6. STATE TRANSITION SYSTEMS*
   - (ω-) automata, games, timed, probabilistic, pushdown
   - programs, Turing machines
   - grammars (Chomsky hierarchy)
7. DECLARATIVE SPECIFICATION
   - Hoare logics, separation logic
   - temporal logics (LTL, CTL, ATL)
   - partial correctness vs. termination, safety vs. liveness

*: Operational.

0.3 Math

Definition 1. *A real b is a <u>bound</u> of a function f from $\mathbb{R}$ to $\mathbb{R}$ if for all x in $\mathbb{R}$, we have $f(x) \leq b$.*

Definition 2. *Given two functions f and g from $\mathbb{R}$ to $\mathbb{R}$, their <u>sum</u> is the function $f+g$ such that for all x in $\mathbb{R}$, we have $(f+g)(x) = f(x)+g(x)$.*

Theorem 1. *For all functions f and g from $\mathbb{R}$ to $\mathbb{R}$, if f and g are bounded, then $f+g$ is bounded.*

Proof.

1. Consider arbitrary functions $\hat{f}$ and $\hat{g}$ from $\mathbb{R}$ to $\mathbb{R}$.
2. Assume $\hat{f}$ and $\hat{g}$ are bounded.
3. Show that $\hat{f}+\hat{g}$ is bounded.
4. ($2 \rightarrow$) Let $\hat{a}$ be a bound for $\hat{f}$, and $\hat{b}$ be a bound for $\hat{g}$.
5. We show that $\hat{a}+\hat{b}$ is a bound for $\hat{f}+\hat{g}$.
6. Consider an arbitrary real $\hat{x}$.
7. Show $(\hat{f}+\hat{g})(x) \leq \hat{a}+\hat{b}$.
8. (Definition of sum) $(\hat{f}+\hat{g})(\hat{x}) = \hat{f}(\hat{x})+\hat{g}(\hat{x})$.
9. (Definition of bound) $\hat{f}(\hat{x}) \leq \hat{a}$ and $\hat{g}(\hat{x}) \leq \hat{b}$.
10. The rest follows from "arithmetic".

$\square$

Homework. Prove the Schröder-Bernstein theorem "in this style".

Definition 3. *Two sets A and B are <u>equipollent</u> ("have the same size") if there is a bijection from A to B.*

Definition 4. *A function f from A to B is*

1. *<u>one-to-one</u> if for all x and y in A, if $x \neq y$, then $f(x) \neq f(y)$.*
2. *<u>onto</u> if for all z in B, there exists x in A such that $f(x) = z$.*

3. *bijective if f is one-to-one and onto.*

Goals	Knowledge	Outermost symbol
Show for all x, $G(x)$. Consider arbitrary $\hat{x}$. Show $G(\hat{x})$.	We know for all x, $K(x)$. In particular, we know $K(\hat{t})$. $\hat{t}$: term containing only constants.	$\forall$
Show there exists x s.t. $G(x)$. We show that $G(\hat{t})$. $\hat{t}$: term containing only constants.	We know there exists x s.t. $K(x)$. Let $\hat{x}$ be s.t. $K(\hat{x})$.	$\exists$

Class 2

Goal	Knowledge	Outermost symbol
Show for all x, $G(x)$. Consider arbitrary $\hat{x}$. Show $G(\hat{x})$	We know for all x, $K(x)$ In particular we know $K(\hat{t})$ for constant $\hat{t}$	$\forall$
Show: exists x s.t. $G(x)$. We show $G(\hat{t})$	We know exists x s.t. $K(x)$ Let $\hat{x}$ be s.t. $K(x)$	$\exists$
Show G_1 iff G_2 1. Show if G_1 then G_2 2. Show if G_2 then G_1	We know K_1 iff K_2 In particular we know if K_1 then K_2 and if K_2 then K_1	$\Longleftrightarrow$
Show if G_1 then G_2 Assume G_1 Show G_2	We know if K_1 then K_2 1. To show K_2 it suffices to show K_2 2. Know K_1, Also know K_2	$\Rightarrow$
Show G_1 and G_2 1. Show G_1 2. Show G_2	Know K_1 and K_2 1. Also Know K_1 2. Also Know K_2	$\wedge$
Show G_1 or G_2 1. Assume $\neg G_1$, show G_2 2. Assume $\neg G_2$, show G_1	We know K_1 or K_2. Show G. 1. Assume K_1, Show G 2. Assume K_2, Show G Case split $\uparrow$	$\vee$
Move Negation Inside, as far as possible		$\neg$

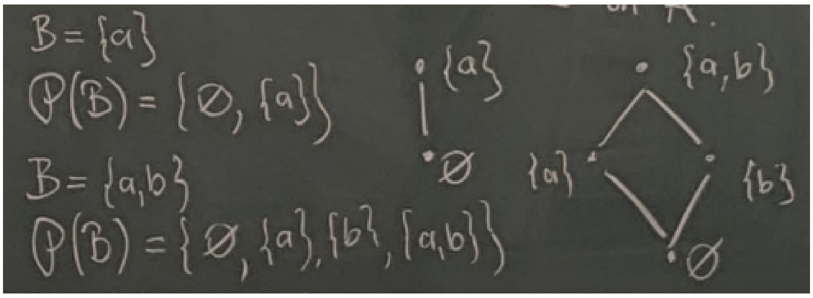
0.4 Lattices and Fixpoints

We begin by defining relations and their properties.

Definition 5. *A binary **relation** R on a set A is a subset $R \subset A \times A$.*
*The relation R is **reflexive** if for all x in A, we have $R(x,x)$.*
*The relation R is **Antisymmetric** if for all x and y in A, if $R(x,y)$ and $R(y,x)$ then $x = y$.*
*The relation R is **transitive** if for all x,y and z in A, if $R(x,y)$ and $R(y,z)$ then $R(x,z)$.*
*The relation R is a **partial order** if R is reflexive, antisymmetric and transitive.*
*A **Poset** $(A,\sqsubseteq)$ is a set A and a partial order $\sqsubseteq$ on A.*

Example 1. *The pair $(\mathbb{N},\leq)$ where $\mathbb{N}$ is the set of natural numbers, is a poset.*
For every set B, we have $(\mathscr{P}(B),\subseteq)$ where $\mathscr{P}(B)$ is the powerset of B, is a poset.

Definition 6.
- *Let $(A,\sqsubseteq)$ be a poset. A function F from A to A is **monotone** (order-preserving, homomorphism) if for all x and y in A, if $x \sqsubseteq y$, then $F(x) \sqsubseteq F(y)$.*
- *F has a fixpoint x in A if there exists x in A such that $F(x) = x$.*



- *x in A is a pre-fixpoint of F if $x \sqsubseteq F(x)$ and is a post-fixpoint of F, if $F(x) \sqsubseteq x$.*

Definition 7. *Let $(A, \sqsubseteq)$ be a poset.*

- *x in A is an **upper bound**(lower bound) on a subset B of A if for all y in B, it holds that $y \sqsubseteq x$ ($x \sqsubseteq y$).*
- *x is the least upper bound of B if (i) x is an upper bound of B and (ii) for all upper bounds y of B, we have $x \sqsubseteq y$. We denote such x by $\bigsqcup B$.*
- *x is the greatest lower bound of B if (i) x is a lower bound of B and (ii) for all lower bounds y of B, we have $y \sqsubseteq x$. We denote such x by $\bigsqcap B$.*

Example 2.
- *Consider the poset $(\mathbb{N}, \leq)$. Then for any $B \subseteq \mathbb{N}$, if B is finite, $\bigsqcup B$ is well-defined and equal to $\max B$. If B is infinite, then $\bigsqcup B$ does not exist.*
- *Consider the poset $(\mathbb{N} \cup \{\infty\}, \leq)$ where for all x in $\mathbb{N}$, it holds that $x \leq \infty$. Then for all $B \subseteq \mathbb{N}$, the least upper bound $\bigsqcup B$ is well-define.*
- *Let A be any set and consider the poset $(\mathscr{P}(A), \subseteq)$. For any subset B of $\mathscr{P}(A)$, it holds that $\bigsqcup B = \bigcup B$ and $\bigsqcap B = \bigcap B$.*

Definition 8. *Poset $(A, \sqsubseteq)$ is a **complete-lattice** if for all $B \subseteq A$, both $\bigsqcap B$ and $\bigsqcup B$ exist.*

Example 3. *Let $(A, \sqsubseteq)$ be a complete-lattice.*

- $\bigsqcup A = \top$
- $\bigsqcap A = \bot$
- $\bigsqcup \varnothing = \bot$
- $\bigsqcap \varnothing = \top$

Theorem 2 (Knaster-Tarski)**.** *For every complete lattice $(A, \sqsubseteq)$ and monotone function F on A, it holds that*

1. *$\bigsqcup \{x \in A | x \sqsubseteq F(x)\}$ is the unique greatest fixpoint of F.*
2. *$\bigsqcap \{x \in A | F(x) \sqsubseteq x\}$ is the unique least fixpoint of F.*

Homework 1. *Prove the Knaster Tarski Theorem.*

Class 3

Definition 9 (Prefixpoint)**.** *Consider a lattice* $(A, \sqsubseteq)$ *and a function* $f\colon A \to A$*. The set of prefixes is*

$$\{x \in A : x \sqsubseteq f(x)\}\,.$$

Definition 10 (Postfixpoint)**.** *Consider a lattice* $(A, \sqsubseteq)$ *and a function* $f\colon A \to A$*. The set of postfixes is*

$$\{x \in A : f(x) \sqsubseteq x\}\,.$$

Definition 11 (gfp and lfp)**.** *Consider a complete lattice* $(A, \sqsubseteq)$ *and a function* $f\colon A \to A$*. Then,*

$$\begin{aligned} \mathrm{gfp} f &:= \bigsqcup \{x \in A : x \sqsubseteq f(x)\} \\ \mathrm{lfp} f &:= \bigsqcap \{x \in A : f(x) \sqsubseteq x\}\,. \end{aligned}$$

Theorem 3 (Fixpoints)**.** *Consider a complete lattice* $(A, \sqsubseteq)$ *and a monotonic function* $f\colon A \to A$*. Then,* $\mathrm{gfp} f$ *and* $\mathrm{lfp} f$ *are fixpoints of* f *and, for all fixpoints* x *of* f*, we have* $\mathrm{lfp} f \sqsubseteq x \sqsubseteq \mathrm{gfp} f$*.*

Definition 12 ($\bigsqcup$-continuous)**.** *Consider a complete lattice* $(A, \sqsubseteq)$*. A function* $f\colon A \to A$ *is* $\bigsqcup$*-continuous if, for all increasing sequences* $x_0 \sqsubseteq x_1 \sqsubseteq x_2 \sqsubseteq x_2 \sqsubseteq \ldots$*, we have*

$$f\left(\bigsqcup \{x_n : n \in \mathbb{N}\}\right) = \bigsqcup \{f(x_n) : n \in \mathbb{N}\}\,.$$

Definition 13 ($\bigsqcap$-continuous)**.** *Consider a complete lattice* $(A, \sqsubseteq)$*. A function* $f\colon A \to A$ *is* $\bigsqcap$*-continuous if, for all increasing sequences* $x_0 \sqsupseteq x_1 \sqsupseteq x_2 \sqsupseteq x_2 \sqsupseteq \ldots$*, we have*

$$f\left(\bigsqcap \{x_n : n \in \mathbb{N}\}\right) = \bigsqcap \{f(x_n) : n \in \mathbb{N}\}\,.$$

Lemma 1. $\bigsqcup$*-continuous implies monotonicity and* $\bigsqcap$*-continuous implies monotonicity.*

Theorem 4 (Constructive fixpoints)**.** *Consider a complete lattice* $(A, \sqsubseteq)$ *and a monotonic function* $f\colon A \to A$*. Then,*

$$\begin{aligned} \mathrm{lfp} f &= \bigsqcup \{f^n(\bot) : n \in \mathbb{N}\} \\ \mathrm{gfp} f &= \bigsqcap \{f^n(\top) : n \in \mathbb{N}\}\,. \end{aligned}$$

Homework 2. *Prove this theorem.*

Definition 14 ($\mathbb{N}$)**.** *Define* $\mathbb{N}$ *as the smallest set* X *such that*

1. $0 \in X$

2. *if $n \in X$, then $Sn \in X$*

In the definition of $\mathbb{N}$, we consider a universal set U sufficiently big, the complete lattice $(2^U, \subseteq)$ and the function on sets given by $f(Y) := \{0\} \cup \{Sn : n \in Y\}$. Then, $\mathrm{lfp} f = \mathbb{N}$.

Definition 15 (Set of words)**.** *Consider a finite alphabet Σ. Define Σ^* as the smallest set X such that*

1. $\varepsilon \in X$
2. *for all $a \in \Sigma$, we have $aX \subseteq X$.*

A few remarks are in place.

- Inductively defined sets are countable and consist of finite elements.
- Inductively defined sets can be written as rules $x \Rightarrow f(x)$ meaning that, if $x \in X$, then $f(x) \in X$.
- Inductively defined sets allow proof by induction. Consider proving that for all $x \in X$ we have $G(x)$. This can be proven by showing
  1. $G(\perp)$
  2. For all $x \in X$, if $G(x)$, then $G(f(x))$

Definition 16 (Balanced binary sequences)**.** *Define the set S as the largest set X such that*

1. $X \subseteq 01X \cup 10X$.

In the definition of balanced binary sequences, we consider the complete lattice $(\Sigma^\omega, \subseteq)$ and the function on sets given by $f(X) := 01X \cup 10X$. Then, balanced binary sequences correspond to $\mathrm{gfp} f$.

Definition 17 (Interval $[0,1]$)**.** *Define the set S as the largest set X such that*

1. $X \subseteq 0X \cup 1X \cup \ldots \cup 9X$.

A few remarks are in place.

- Coinductively defined sets are uncountable and consist of infinite elements.
- Coinductively defined sets can be written as rules $x \Leftarrow f(x)$ meaning that, for all $y \in X$, there exists x such that $y = f(x)$ and $x \in X$.
- Coinductively defined sets allow proof by coinduction. Consider proving that for all x, if $G(x)$, then $x \in X$. This can be proven by showing
  1. For all x and i, if $G(f_i(x))$, then $G(x)$,

  where $\{f_1, \ldots, f_n\}$ is the set of rules that define the set X.

Homework 3 (Prove balanced binary sequences)**.** *Consider S generated by the rules $X \Leftarrow 01X$ and $X \Leftarrow 10X$. Prove that, for all binary words x, we have that x inS if and only if every finite prefix of even length of x has the same number of 0s and 1s.*

Hints.

1. The direction $\Leftarrow$ can be proven by coinduction.
2. The direction $\Rightarrow$ can be proven by induction on the length of the prefix.

Class 6

Formal system F is a set of rules. Rule is a finite set of (formulas) premises $p_0, \dots, p_k$ and (a formula called) conclusion c. We usually have infinitely many rules but only finitely many different rule schemata. For example, schema $\phi \to \phi$ gives infinitely many rules like $p_3 \to p_3$. Axiom is a rule without premises.

Proof (derivation) is a finite sequence of formulas $\phi_0, \dots, \phi_n$ such that every formula in the sequence is

- either an axiom (which can be viewed as a special case of the following);
- or the conclusion of a rule whose premises occur earlier in the sequence.

This is a linear view.

Linear view is usually easier for proving meta theorems. Tree view (inductive definition) is usually better in practice.

Theorem is a formula that occurs in a proof. We distinguish the following:

- $\vdash \phi$ ... "ϕ is a theorem (of the formal system F)" (has a proof) [syntax]
- $\vDash \phi$ ... "ϕ is valid (ϕ is tautology)" (is true in all models) [semantics]

Formal system equipped with semantics is called a logic. Most of logic is about establishing $\vdash \phi$ iff $\vDash \phi$.

Rule R is sound iff [if all premises of R are valid, then the conclusion of R is valid]. Formal system F is sound iff all rules are sound (or equivalently, every theorem is valid). Formal system F is complete iff every valid formula is a theorem. Formal system F is consistent unless $\vdash \bot$ (or equivalently, there exists a formula that is not a theorem). Rule R is derivable in F iff [for all formulas ϕ, $\underset{F \cup \{R\}}{\vdash} \phi$ iff $\underset{F}{\vdash} \phi$]. Rule R is admissible in F iff $F \cup \{R\}$ is still consistent. Formula ϕ is expressible in a logic L iff [there exists a formula ψ of L such that, for all interpretations v, $[[\phi]]_v = [[\psi]]_v$. For example $\phi_1 \wedge \phi_2$ is expressible using only $\neg$ and $\vee$ (de Morgan) as $\psi = \neg(\neg\phi_1 \wedge \neg\phi_2)$.

We can enumerate all theorems by systematically enumerating all possible proofs. The proof is a witness for validity. Sound formal system gives a sound procedure for validity (but not necessarily complete). Sound complete formal system gives a sound semi-complete procedure for validity (may not terminate on inputs that represent a formula that is not valid). To get a decision procedure (sound and complete procedure for validity), we need both (1) sound complete formal system for validity, and (2) sound complete formal system for satisfiability (to define a formal system for satisfiability, replace "formulas" (ϕ is valid) by "judgements" (ϕ is satisfiable); all axioms are satisfiable, all rules go from satisfiables to satisfiable). For every input ϕ, one of them will eventually terminate. Conclude; either ϕ is valid, or $\neg\phi$ is satisfiable (which means that ϕ is not valid). Recall that, if both a set and its complement are recursively-enumerable, the set is recursive (decidable).

Example (formal system for unsatisfiability):

$$\frac{\Gamma[\bot] \qquad\qquad \Gamma[\top]}{\Gamma[p]}$$

0.5 Hilbert formal system for propositional logic

Hilbert system uses connectives $\rightarrow$ and $\neg$ only. Hilbert system has three axioms and one rule – modus ponens (MP):

$$\frac{\phi \qquad\qquad \phi \rightarrow \psi}{\psi}$$

Axioms:

- (K): $\phi \rightarrow \psi \rightarrow \phi$
- (S): $(\phi \rightarrow \psi \rightarrow \chi) \rightarrow ((\phi \rightarrow \psi) \rightarrow (\phi \rightarrow \chi))$
- (EM): $(\neg\phi \rightarrow \neg\psi) \rightarrow (\psi \rightarrow \phi)$

TODO