Probabilistic Programming, excercises 3. Jan Boerman, Jasper van Rooijen.

gratis collegeblok

Jan Boerman
Jasper van Rooijen

Onderwerp:
Datum:

1a The domain of natural numbers is an infinite set with no largest element according to the less-than-or-equals relation. ~~an a~~ Infinity could be an upper bound but it is not part of the domain.
Therefore, $(\mathbb{N}, \leq)$ is not a complet lattice

b ~~Yes the domain of complet lattices~~
~~every (D, ⊑~~
Yes. every $(D, \sqsupseteq)$ is a complete lattice if their reverse-ordered counterpart $(D, \sqsubseteq)$ is a complete lattice. ~~This~~
Reflexivity, transitivity, and antisymmetry are still maintained, but the reverse order implies that the supremum becomes the infimum and vice versa.

c. ~~Yes this is a complet lattice.~~
No, this is not a complete lattice as this is not even a partial order.
Not every element in the domain can be compared with another. e.g. $(10, 20) \sqsubseteq (20, 10)$

gratis collegeblok

Jan Boerman
Jasper van Rooijen

Onderwerp:

Datum:

2a show that $wp[c](0) = 0$.

proof by structural induction over the structure of c.

Base cases

① skip. $wp[skip](0) = 0$
by definition

② diverge $wp[diverge](0) = 0$
because it's always 0 for a diverge, by definition

③ assignment. $wp[x := E](0) = 0[x := E]$
$= 0$

Induction Step: suppose we already know that for programs p and q
$wp[p](0) = 0$ and $wp[q](0) = 0$.

Then:

④ $wp[p;q](0) = wp[p](wp[q](0))$
$= wp[p](0)$ (apply induction hypothesis)
$= 0$ (apply induction hypothesis)

⑤ $wp[\text{if}(G)\ P \text{ else } Q](0) = (G \wedge wp[P](0)) \vee (\neg G \wedge wp[q](0))$
$= (G \wedge 0) \vee (\neg G \wedge 0)$ (apply IH twice)

gratis collegeblok

Onderwerp:

Datum:

⑥ $wp[\text{while}(G)\,P](0) =$

$\quad lfp\ X.\ ((G \wedge wp[P](X)) \vee (\neg G \wedge 0)$

$\quad = lfp\ X.\ ((G \wedge wp[P](X))$

apply kleene's fixpoint theorem

$\Phi(X) = G \wedge wp[P](X)$

$lfp\ \Phi = \sup_{n \in \mathbb{N}} \Phi^n(\text{false})$

let's try $n=0$. we get

$\Phi(0) = G \wedge wp[P](0)$

$\quad = G \wedge 0$ (apply IH)

$\quad = 0.$

Since there can be no smaller set than 0, this is the supremum. The while-case is therefore proven at last.

The combination of base cases ①, ②, ③ and induction cases ④, ⑤, ⑥ proves that for any arbitrairy program C $wp[C](0) = 0$

QED.

gratis collegeblok

Onderwerp:

Datum:

assume we know $f \to g$, we must show $wp[c](f) \to wp[c](g)$

2b proof by structural induction

base cases

① $wp[skip](f) \Rightarrow wp[skip](g)$

$f \Rightarrow g$ QED 1/6

② $wp[diverge](f) \Rightarrow wp[diverge](g)$

$0 \Rightarrow 0$ QED 2/6

③ $wp[x:=E](f) \Rightarrow wp[x:=E](g)$

$f[x:=E] \Rightarrow g[x:=E]$ QED 3/6

Induction Step: suppose we already know for programs p and q

$(f \to g) \to (wp[p](f) \to wp[p](g))$

and $(f \to g) \to (wp[q](f) \to wp[q](g))$

④ $wp[p;q](f) \to wp[p;q](g)$

$wp[p](wp[q](f)) \to wp[p](wp[q](g))$

$wp[p](wp[q](g)) \to wp[p](wp[q](g))$

(strengthen assumption, apply IH)

QED 4/6

⑤ $wp[if(G)\ then\ p\ else\ q](f) \to wp[if(G)\ then\ p\ else\ q](g)$

$((G \wedge wp[p](f)) \vee (\neg G \wedge wp[q](f))$

$\to$

$((G \wedge wp[p](g)) \vee (\neg G \wedge wp[q](g)))$

just apply IH and we get a formula in the form of $a \to a$ ~~just the~~ similarly to ④ QED 5/6

gratis collegeblok

Onderwerp:

Datum:

(6) $wp[while(G)\ p](f) \rightarrow wp[while(G)\ p](g)$

$lfp\ X. ((G \wedge wp[p](X)) \vee (\neg G \wedge f)$

$\overrightarrow{lfp\ Y. ((G \wedge wp[p](Y)) \vee (\neg G \wedge g)}$

~~Since we know $(f \rightarrow g)$ and implication distributes over disjunction~~

$(\neg G \wedge f) \rightarrow (\neg G \wedge g)$

this is trivially true since implication distributes over conjunction. QED 6/6

By proving ~~all~~ the statement true for all possible structures of C, the statement is true for any arbitrary C.

2c Yes. The predicates f and g can be seen as ~~assignmen~~ sets of assignement to variables in the program.

$f \vee g$ means a union of those sets. It does not matter we take the wp first and then the union, or ~~$wp(f) \vee wp(g)$~~ first the union and then the wp.

~~In both~~ In both cases, the same set of variables will still be "tracked".

gratis collegeblok

Jan Boerman
Jasper van Rooijen

Onderwerp:
Datum:

3 a $(y > 0) \rightarrow (x == z \cdot y)$

b $((G \wedge (y > 0) \rightarrow (x == z \cdot y)) \rightarrow$
$wp[z := z+1; x := x-y]((y>0) \rightarrow (x == z \cdot y))$
$= (G \wedge (y>0) \rightarrow (x == z \cdot y)) \rightarrow$
$wp[z := z+1](wp[x := x-y]((y>0 \rightarrow (x == z \cdot y)))$
$= (y \leq 0) \vee ((G \wedge (x == z \cdot y)) \rightarrow$
$wp[z := z+1](wp[x := x-y](x == z \cdot y)))$
$= y \leq 0 \vee \left(\begin{array}{l} (G \wedge (x == y \cdot z)) \rightarrow \\ wp[z := z+1](x == z \cdot y[x := x-y]) \end{array} \right)$
$= y \leq 0 \vee \left(\begin{array}{l} (G \wedge (x == z \cdot y)) \rightarrow \\ wp[z := z+1](x-y == z \cdot y) \end{array} \right)$
$= y \leq 0 \vee \left(\begin{array}{l} (G \wedge (x == z \cdot y)) \rightarrow \\ (x-y == z \cdot y)[z := z+1] \end{array} \right)$
$= y \leq 0 \vee \left(\begin{array}{l} (G \wedge (x == z \cdot y)) \rightarrow \\ (x-y == (z+1) \cdot y) \end{array} \right)$
$= y \leq 0 \vee \left(\begin{array}{l} (G \wedge (x == z \cdot y)) \rightarrow \\ (x-y == z \cdot y + y) \end{array} \right)$
$= y \leq 0 \vee ((G \wedge (x == z \cdot y)) \rightarrow (x-2y == z \cdot y))$

~~there does not seem to~~

$\cancel{y \leq 0 \vee (G \wedge x == x - 2y}$

there does not seem to be a solution other than $y \leq 0$.

gratis collegeblok

Jan Boerman
Jasper van Rooijen

Onderwerp:
Datum:

4 $\phi(x) = (G \wedge wp(P, x)) \vee (\neg G \wedge F)$

ϕ is continuous if for every non-empty chain $S \subseteq \mathbb{P}$
$\phi(\bigsqcup S) = \bigsqcup \phi(S)$

We know $(\mathbb{P}, \sqsubseteq)$ is a complete lattice thus ever subset $X \subseteq P$ has a least upper bound

let's try to compute. ($\mathbb{P}$ is the domain of predicates)

$\phi(\bigsqcup S) = \bigsqcup \phi(S) \iff$

$(G \wedge wp(P, \bigsqcup S)) \vee (\neg G \wedge F) =$
$\bigsqcup((G \wedge wp(P, S)) \vee (\neg G \wedge F))$
$\iff$
$(G \wedge \bigsqcup S) \vee (\neg G \wedge F) =$
$\bigsqcup((G \wedge \bigsqcup S) \vee (\neg G \wedge F)$

This is trivially true.

gratis collegeblok

Jan Boerman
Jasper van Rooijen

Onderwerp:

Datum:

5 a we need to show that $(D \to D, \subseteq)$
is ① Reflexive, ② transitive and ③ antisymetric

D is the domain of sets

$D \to D$ is a ~~[illegible]~~ set of functions of type set to set.

proof of ① $f(d) \subseteq f(d)$

true by definition of the subset relation any set is its own subset.

proof of ② $f(d) \subseteq g(d) \wedge g(d) \subseteq h(d) \Rightarrow f(d) \subseteq h(d)$

for ~~all~~ $d \in D$ and any $f, g, h \in D \to D$

also true by definition of the subset relation

proof of ③ $f(d) \subseteq g(d) \wedge g(d) \subseteq f(d) \Rightarrow f(d) = g(d)$

true by deffinition of set equivalence

if set A contains all elements of set B and set B contains all elements of set A, then set A and B are equivalent.

gratis collegeblok

Onderwerp:

Datum:

5b to show $(D \to D, \sqsubseteq)$ is a complete lattice we need to show that

① $(D \to D, \sqsubseteq)$ is a partial order and

② all subsets of $D \to D$ have a supremum and

③ all subsets of $D \to D$ have an infimum

proof of ①: proven in 5a

" " ②:

The Supremum of a subset is the function that returns the intersection of all sets returned by other funtions in the subset that the same parameter.

$S \subseteq D \to D$

$Sup(S) = f(d) \mapsto \bigcap_{g \in S} \{g(d)\}$

where $d \in D$, $f \in S$.

proof of ③: similar as ②, but using a union

~~$inf(S) = \bigcup_{g \in S} g(d)$~~

$inf(S) = f(d) \mapsto \bigcup_{g \in S} g(d)$

where $d \in D$, $f \in S$.