Incremental Fixpoint Computation

Lambert Meertens*
Department of Algorithmics and Architecture, CWI, Amsterdam, and
Department of Computing Science, Utrecht University, The Netherlands
http://www.cwi.nl/cwi/people/Lambert.Meertens.html

Printed April 11, 2000

1 In the following $(A, \sqsupseteq)$ is some poset, and $F : A \to A$ denotes a monotonic function.

2 A value $x \in A$ is a *prefixpoint* of F when $x \sqsupseteq Fx$.

3 Define $\top\!\!\!\top = \sqcap(\text{false}^{\mathsf{K}} : \mathsf{id})$ and $\bot\!\!\!\bot = \sqcap\,\mathsf{id}$. If A is $\sqcap$-complete, both exist. We show that $\top\!\!\!\top$ and $\bot\!\!\!\bot$ — in spite of the asymmetric definitions given — are each other's dual, i.e., $\top\!\!\!\top$ is synonymous with $\top\!\!\!\top' = \sqcup\,\mathsf{id}$.

Proof. We show that both $\top\!\!\!\top$ and $\top\!\!\!\top'$ dominate all elements of A.

First,

$$
\begin{array}{ll}
 & \top\!\!\!\top \sqsupseteq z \\
\equiv & \qquad \{\text{definition of } \top\!\!\!\top\} \\
 & \sqcap(\text{false}^{\mathsf{K}} : \mathsf{id}) \sqsupseteq z \\
\equiv & \qquad \{\sqcap\text{-characterization}\} \\
 & \forall(\text{false}^{\mathsf{K}} : \mathsf{id} \mathrel{\dot{\sqsupseteq}} z) \\
\equiv & \qquad \{\text{shunting, propositional calculus}\}
\end{array}
$$

*This work was performed while visiting Kestrel Institute, Palo Alto.

$\quad$ true

Next,

$$
\begin{array}{ll}
 & \top\!\!\top' \sqsupseteq z \\
\equiv & \quad \{\text{definition of } \top\!\!\top'\} \\
 & \bigsqcup \mathsf{id} \sqsupseteq z \\
\Leftarrow & \quad \{\bigsqcup\text{-instantiation}\} \\
 & \text{true}
\end{array}
$$

So

$$
\begin{array}{ll}
 & \top\!\!\top = \top\!\!\top' \\
\equiv & \quad \{\text{indirect equality}\} \\
 & \forall(z :: \top\!\!\top \sqsupseteq z \;\equiv\; \top\!\!\top' \sqsupseteq z) \\
\equiv & \quad \{\text{just shown}\} \\
 & \forall(z :: \text{true} \;\equiv\; \text{true}) \\
\equiv & \quad \{\text{propositional calculus}\} \\
 & \text{true}
\end{array}
$$

End of proof.

4 If A is finite and $\sqcap$-complete (i.e., all binary $\sqcap$s exist), then all non-empty $\bigsqcap$s exist. If, moreover, $\top\!\!\top$ exists, A is $\bigsqcap$-complete.

5 If A is $\bigsqcap$-complete, it is also μ-complete, i.e., μF exists and equals $\bigsqcap(\mathsf{id} \mathrel{\dot\sqsupseteq} F : \mathsf{id})$. In words, the least fixpoint of F is the infimum of the set of prefixpoints of F.

6 A stream x is a mapping $x : \mathbb{N} \to B$ for some B. Instead of $x.n$ we also write, equivalently, x_n.

7 A stream x is *finite* when there exists a value x_∞ and a natural s such that $\forall(n :: x_{s+n} = x_\infty)$, and we say then that x is *finished at* s and has *final value* x_∞. If such an x_∞ and s exist, x_∞ is unique but s is not.

If x is finished at s, $x_\infty = x_s$. It follows that x is finite iff there exists an s such that $\forall(n :: x_{s+n} = x_s)$, or, equivalently, $\forall(n :: x_{s+n+1} = x_{s+n})$.

8 A stream $x : \mathbb{N} \to A$ is called *ascending* when x is a monotonic mapping, that is, $x_i \sqsupseteq x_j \;\Leftarrow\; i \geq j$.

If A is finite, each ascending stream is — non-constructively — finite.

Defining $\overline{x} = (n :: \bigsqcup(i : i \leq n : x_i))$, the stream $\overline{x}$ — if all $\bigsqcap$s involved exist — is ascending for all x. Moreover, x is ascending iff $\overline{x} = x$.

9 A finite ascending stream x has final value $x_\infty = \bigsqcup x$.

10

Theorem 1: Let x be an ascending stream satisfying

(i) $\quad F\,x_n \;\sqsupseteq\; x_{n+1}$ for all n

(ii) $\quad x_0 \;=\; \perp\!\!\!\perp$

(iii) $\quad x_{n+1} \;\sqsupset\; x_n \;\vee\; x_n \;\sqsupseteq\; F x_n$ for all n

Then:

(a) if there exists a natural s such that $x_s \sqsupseteq F x_s$, then x is finite

(b) $\mu F \;\sqsupseteq\; x_n$ for all n

(c) if x is finite, $x_\infty \;=\; \mu F$

Proof. Denote $P \;=\; \mathsf{id} \mathrel{\dot{\sqsupseteq}} F$, i.e., the values satisfying P are the prefixpoints of F.

For part (a), we only need — next to the ascent of x — assumption (i). Define $Q_n \;\equiv\; x_{n+1} = x_n \wedge P x_n$. We will show that Q is ascending, but first we show that $Q = P \circ x$:

$$
\begin{array}{ll}
 & Q_n \;\equiv\; P x_n \\
\equiv & \qquad \{\text{definition of } Q\} \\
 & (x_{n+1} = x_n \wedge P x_n) \;\equiv\; P x_n \\
\equiv & \qquad \{\text{propositional calculus}\} \\
 & x_{n+1} = x_n \;\Leftarrow\; P x_n
\end{array}
$$

$$
\begin{array}{ll}
\equiv & \quad \{x_{n+1} \sqsupseteq x_n,\ \sqsupseteq \text{ is antisymmetric}\} \\
& x_n \sqsupseteq x_{n+1} \ \Leftarrow\ Px_n \\
\equiv & \quad \{\text{definition of } P\} \\
& x_n \sqsupseteq x_{n+1} \ \Leftarrow\ x_n \sqsupseteq Fx_n \\
\equiv & \quad \{(i)\ \ Fx_n \sqsupseteq x_{n+1},\ \sqsupseteq \text{ is transitive}\} \\
& \text{true}
\end{array}
$$

Then Q is ascending, since:

$$
\begin{array}{ll}
& Q_{n+1} \ \Leftarrow\ Q_n \\
\equiv & \quad \{Q = P \circ x\} \\
& Px_{n+1} \ \Leftarrow\ Q_n \\
\equiv & \quad \{\text{definition of } Q\} \\
& Px_{n+1} \ \Leftarrow\ (x_{n+1} = x_n \wedge Px_n) \\
\equiv & \quad \{\text{equational logic}\} \\
& \text{true}
\end{array}
$$

We are now ready to show that x finishes at s if s is such that $x_s \sqsupseteq Fx_s$:

$$
\begin{array}{ll}
& \forall(n :: x_{s+n+1} = x_{s+n}) \\
\Leftarrow & \quad \{\text{definition of } Q\} \\
& \forall(n :: Q_{s+n}) \\
\equiv & \quad \{Q \text{ is ascending}\} \\
& \forall(n :: \overline{Q}_{s+n}) \\
\equiv & \quad \{\text{definition of } \overline{Q}\} \\
& \forall(n :: \exists(i : i \leq s+n : Q_i)) \\
\Leftarrow & \quad \{\exists\text{-instantiation}\} \\
& Q_s \\
\equiv & \quad \{Q = P \circ x\} \\
& Px_s \\
\equiv & \quad \{\text{definition of } P\} \\
& x_s \sqsupseteq Fx_s
\end{array}
$$

For part (b) we use, in addition, assumption (ii), which provides the basis of a proof by natural induction. For the step:

$$
\begin{array}{ll}
 & \mu F \sqsupseteq x_{n+1} \\
\Leftarrow & \qquad \{(i) \;\; Fx_n \sqsupseteq x_{n+1},\ \sqsupseteq \text{ is transitive}\} \\
 & \mu F \sqsupseteq Fx_n \\
\equiv & \qquad \{\mu F \text{ is fixpoint}\} \\
 & F\mu F \sqsupseteq Fx_n \\
\Leftarrow & \qquad \{F \text{ is monotonic}\} \\
 & \mu F \sqsupseteq x_n
\end{array}
$$

For part (c) we also use assumption (iii). Assume x finishes at s with final value x_∞. Instantiating $n = s$ in (iii), and using $x_{s+1} = x_s = x_\infty$, we obtain

$$x_\infty \sqsupset x_\infty \;\vee\; x_\infty \sqsupseteq Fx_\infty$$

which by the antisymmetry of $\sqsupseteq$ simplifies to

$$x_\infty \sqsupseteq Fx_\infty$$

Then

$$
\begin{array}{ll}
 & x_\infty = \mu F \\
\equiv & \qquad \{\text{fixpoint properties}\} \\
 & Fx_\infty = x_\infty \;\wedge\; \mu F \sqsupseteq x_\infty \\
\equiv & \qquad \{x_\infty \sqsupseteq Fx_\infty,\ \sqsupseteq \text{ is antisymmetic}\} \\
 & Fx_\infty \sqsupseteq x_\infty \;\wedge\; \mu F \sqsupseteq x_\infty \\
\equiv & \qquad \{x_\infty = x_s = x_{s+1}\} \\
 & Fx_s \sqsupseteq x_{s+1} \;\wedge\; \mu F \sqsupseteq x_s \\
\equiv & \qquad \{\text{left conjunct: } (i)\text{; right conjunct: } (b)\} \\
 & \text{true}
\end{array}
$$

End of proof.

11 Call $(A, \sqsupseteq)$ well-roofed if each ascending stream is finite. A sufficient condition is finiteness of A.

The Theorem of item **10** gives a way to compute least fixpoints in well-roofed posets.

Assume F to be given. Let $\mathcal{P}$ be any procedure — possibly non-deterministic, but effective — that, for given input x_n, produces output value x_{n+1} satisfying:

$$F\,x_n \;\sqsupseteq\; x_{n+1} \;\sqsupseteq\; x_n$$

$$Fx_n \;=\; x_n \;\Leftarrow\; x_{n+1} \;=\; x_n$$

So output x_{n+1} is bounded between Fx_n and x_n, and may only equal x_n if x_n is a fixpoint.

Then any stream starting with $x_0 = \perp\!\!\!\perp$ and generated by iterating $\mathcal{P}$ for $n = 0, 1, 2, \ldots$, satisfies the conditions of the Theorem.

If the stream finishes — which is guaranteed under the assumption of well-roofedness — its final value is the least fixpoint. Otherwise, an (infinite) strictly ascending stream is produced.

12 A simple procedure is: take $x_{n+1} = Fx_n$. To see that x is ascending, we appeal to induction.

(Basis)

$$
\begin{array}{ll}
 & x_1 \;\sqsupseteq\; x_0 \\
\equiv & \qquad \{\text{definition of } x_0\} \\
 & x_1 \perp\!\!\!\perp \;\sqsupseteq\; \perp\!\!\!\perp \\
\equiv & \qquad \{\perp\!\!\!\perp\text{-characterization}\} \\
 & \text{true}
\end{array}
$$

(Step)

$$
\begin{array}{ll}
 & x_{n+2} \;\sqsupseteq\; x_{n+1} \\
\equiv & \qquad \{\text{definition of } x\} \\
 & Fx_{n+1} \;\sqsupseteq\; Fx_n \\
\Leftarrow & \qquad \{F \text{ is monotonic}\} \\
 & x_{n+1} \;\sqsupseteq\; x_n
\end{array}
$$

13 Given two posets $(A, \sqsupseteq_A)$ and $(B, \sqsupseteq_B)$, the product ordering $\sqsupseteq_A \times \sqsupseteq_B$, denoted below by $\sqsupseteq_\times$, is a relation on $A \times B$ defined by

$$(a_0, b_0) \sqsupseteq_\times (a_1, b_1) \quad \equiv \quad a_0 \sqsupseteq_A a_1 \;\wedge\; b_0 \sqsupseteq_B b_1$$

It is again a partial-order relation.

Proof. In the proof expressions we omit the subscripts $_A$ and $_B$ since they can be immediately reconstructed and play no essential role.

(Reflexive antisymmetry) We combine the conjunction of the reflexivity and (weak) antisymmetry laws of relation R into the single *reflexive-antisymmetry* law $xRy \wedge yRx \;\equiv\; x = y$.

$$
\begin{array}{ll}
 & (a_0, b_0) \sqsupseteq_\times (a_1, b_1) \;\wedge\; (a_1, b_1) \sqsupseteq_\times (a_0, b_0) \\
\equiv & \qquad \{\text{definition of } \sqsupseteq_\times\} \\
 & a_0 \sqsupseteq a_1 \;\wedge\; b_0 \sqsupseteq b_1 \;\wedge\; a_1 \sqsupseteq a_0 \;\wedge\; b_1 \sqsupseteq b_0 \\
\equiv & \qquad \{\text{reshuffling, reflexive-antisymmetry of } \sqsupseteq_A \text{ and } \sqsupseteq_B\} \\
 & a_0 = a_1 \;\wedge\; b_0 = b_1 \\
\equiv & \qquad \{\text{equality of pairs}\} \\
 & (a_0, b_0) \;=\; (a_1, b_1)
\end{array}
$$

(Transitivity)

$$
\begin{array}{ll}
 & (a_0, b_0) \sqsupseteq_\times (a_2, b_2) \\
\equiv & \qquad \{\text{definition of } \sqsupseteq_\times\} \\
 & a_0 \sqsupseteq a_2 \;\wedge\; b_0 \sqsupseteq b_2 \\
\Leftarrow & \qquad \{\text{transitivity of } \sqsupseteq_A \text{ and } \sqsupseteq_B\} \\
 & a_0 \sqsupseteq a_1 \;\wedge\; a_1 \sqsupseteq a_2 \;\wedge\; b_0 \sqsupseteq_B b_1 \;\wedge\; b_1 \sqsupseteq_B b_2 \\
\equiv & \qquad \{\text{reshuffling, definition of } \sqsupseteq_\times\} \\
 & (a_0, b_0) \sqsupseteq_\times (a_1, b_1) \;\wedge\; (a_1, b_1) \sqsupseteq_\times (a_2, b_2)
\end{array}
$$

End of proof.

14 The binary partial-order product can be generalized to the product of any indexed collection of partial orders, at the same time generalizing lifted relation $\dot{\sqsupseteq}$.

To define it we use a notation for "I-tuples", where I is the index set, that generalizes function comprehension. Let $\forall(i : i{\in}I : a_i \in A_i)$. Then $(i : i{\in}I : a_i)$ denotes the corresponding element of $\prod(i : i{\in}I : A_i)$.

Let a poset $(A_i,\ \sqsupseteq_i)$ be given for each $i \in I$. Then $\prod(i : i{\in}I : \sqsupseteq_i)$, denoted below by $\sqsupseteq_\Pi$, is a partial-order relation on $\prod(i : i{\in}I : A_i)$ defined by

$$(i : i{\in}I : a_i) \sqsupseteq_\Pi (i : i{\in}I : b_i) \quad \equiv \quad \forall(i : i{\in}I : a_i \sqsupseteq_i b_i)$$

The proof that this gives a partial order runs along the same lines as the proof just given for the binary version.

15 If $(i : i{\in}I : a_i) \sqsupset_\Pi (i : i{\in}I : b_i)$, then there is some $i \in I$ such that $a_i \sqsupset_i b_i$.

Proof.

$$\begin{array}{ll}
 & \exists(i : i{\in}I : a_i \sqsupset_i b_i) \\
\equiv & \qquad \{\text{definition of } \sqsupset\} \\
 & \exists(i : i{\in}I : a_i \sqsupseteq_i b_i \ \wedge\ a_i \neq b_i) \\
\Leftarrow & \qquad \{\forall\text{-instantiation}\} \\
 & \exists(i : i{\in}I : \forall(i : i{\in}I : a_i \sqsupseteq_i b_i) \ \wedge\ a_i \neq b_i) \\
\equiv & \qquad \{\wedge\text{-}\exists\text{-distribution}\} \\
 & \forall(i : i{\in}I : a_i \sqsupseteq_i b_i) \ \wedge\ \exists(i : i{\in}I : a_i \neq b_i) \\
\equiv & \qquad \{\text{definition of } \sqsupseteq_\Pi\text{, equality of tuples}\} \\
 & (i : i{\in}I : a_i) \sqsupseteq_\Pi (i : i{\in}I : b_i) \ \wedge\ (i : i{\in}I : a_i) \neq (i : i{\in}I : b_i) \\
\equiv & \qquad \{\text{definition of } \sqsupset\} \\
 & (i : i{\in}I : a_i) \sqsupset_\Pi (i : i{\in}I : b_i)
\end{array}$$

End of proof.

16 Given two posets $(A,\ \sqsupseteq_A)$ and $(B,\ \sqsupseteq_B)$, the lexical ordering $\sqsupseteq_A \rtimes \sqsupseteq_B$, denoted below by $\sqsupseteq_\rtimes$, is a relation on $A{\times}B$ defined by

$$(a_0, b_0) \sqsupseteq_\rtimes (a_1, b_1) \quad \equiv \quad a_0 \sqsupseteq_A a_1 \ \wedge\ (a_0 \sqsupset_A a_1 \ \vee\ b_0 \sqsupseteq_B b_1)$$

It is again a partial-order relation, and a weakening of the product ordering.

Proof. (Reflexive antisymmetry)

$$\begin{array}{ll}
 & (a_0, b_0) \sqsupseteq_\rtimes (a_1, b_1) \ \wedge\ (a_1, b_1) \sqsupseteq_\rtimes (a_0, b_0) \\
\equiv & \qquad \{\text{definition of } \sqsupseteq_\rtimes\}
\end{array}$$

$$a_0 \sqsupseteq a_1 \;\wedge\; (a_0 \sqsupset a_1 \;\vee\; b_0 \sqsupseteq b_1) \wedge$$
$$a_1 \sqsupseteq a_0 \;\wedge\; (a_1 \sqsupset a_0 \;\vee\; b_1 \sqsupseteq b_0)$$
$\equiv$ {reshuffling, reflexive-antisymmetry of $\sqsupseteq_A$ }
$$a_0 = a_1 \;\wedge\; (a_0 \sqsupset a_1 \;\vee\; b_0 \sqsupseteq b_1) \;\wedge\; (a_1 \sqsupset a_0 \;\vee\; b_1 \sqsupseteq b_0)$$
$\equiv$ {$\wedge$-$\vee$-distribution, strong antisymmetry of $\sqsupset_A$ }
$$a_0 = a_1 \;\wedge\; b_0 \sqsupseteq b_1 \;\wedge\; b_1 \sqsupseteq b_0)$$
$\equiv$ {reflexive-antisymmetry of $\sqsupseteq_B$ }
$$a_0 = a_1 \;\wedge\; b_0 = b_1$$
$\equiv$ {equality of pairs}
$$(a_0, b_0) \;=\; (a_1, b_1)$$

(Transitivity)

$$(a_0, b_0) \sqsupseteq_{\rtimes} (a_2, b_2)$$
$\equiv$ {definition of $\sqsupseteq_{\rtimes}$ }
$$a_0 \sqsupseteq a_2 \;\wedge\; (a_0 \sqsupset a_2 \;\vee\; b_0 \sqsupseteq b_2)$$
$\Leftarrow$ {order properties}
$$a_0 \sqsupseteq a_1 \wedge a_1 \sqsupseteq a_2 \wedge$$
$$((a_0 \sqsupseteq a_1 \wedge a_1 \sqsupset a_2) \vee (a_0 \sqsupset a_1 \wedge a_1 \sqsupseteq a_2) \vee$$
$$(b_0 \sqsupseteq b_1 \wedge b_1 \sqsupseteq b_2))$$
$\Leftarrow$ {propositional calculus}
$$a_0 \sqsupseteq a_1 \;\wedge\; (a_0 \sqsupset a_1 \;\vee\; b_0 \sqsupseteq b_1) \wedge$$
$$a_1 \sqsupseteq a_2 \;\wedge\; (a_1 \sqsupset a_2 \;\vee\; b_1 \sqsupseteq b_2)$$
$\equiv$ {definition of $\sqsupseteq_{\rtimes}$ }
$$(a_0, b_0) \sqsupseteq_{\rtimes} (a_1, b_1) \;\wedge\; (a_1, b_1) \sqsupseteq_{\rtimes} (a_2, b_2)$$

(Weakening)

$$(a_0, b_0) \sqsupseteq_{\rtimes} (a_1, b_1)$$
$\equiv$ {definition of $\sqsupseteq_{\rtimes}$ }
$$a_0 \sqsupseteq a_1 \;\wedge\; (a_0 \sqsupset a_1 \;\vee\; b_0 \sqsupseteq b_1)$$
$\Leftarrow$ {propositional calculus}
$$a_0 \sqsupseteq a_1 \;\wedge\; b_0 \sqsupseteq b_1$$
$\equiv$ {definition of $\sqsupseteq_{\times}$ }

$$(a_0, b_0) \sqsupseteq_\times (a_1, b_1)$$

End of proof.

17 Let $(A, \sqsupseteq) = (\prod(i : i{\in}I : A_i), \prod(i : i{\in}I : \sqsupseteq_i))$, where $(A_i, \sqsupseteq_i)$ is a poset for all $i \in I$. Below we omit the subscripts on the order relations $\sqsupseteq_i$.

To select the element indexed by i from I-tuple $x \in \prod(i : i{\in}I : A_i)$ we write $x.i$, so

$$x = (i : i{\in}I : a_i) \quad \equiv \quad \forall(i : i{\in}I : x.i = a_i)$$

The *tuple-update* notation $x[.j{\mapsto}u]$, for $j \in J, u \in A_j$ is then defined by:

$$\forall(i : i{\in}I \wedge i \neq j : x[.j{\mapsto}u].i = x.i) \;\wedge\; x[.j{\mapsto}u].j = u$$

Let further $F : A \to A$ be a monotonic function, and assume A is well-roofed.

We give a procedure as in **11** for the iterative computation of μF.

Given input x_n, output x_{n+1} is computed non-deterministically as follows:

Putting $y_n = F.x_n$,

(Case A) $\exists(j : j{\in}I : y_n.j \sqsupset x_n.j) :$ $x_{n+1} = x_n[.j{\mapsto}y_n.j]$

(Case B) otherwise : $x_{n+1} = x_n$

The procedure is non-deterministic by its freedom to pick j. Note that, possibly, not all components of the I-tuple y_n have to be computed, but only as many as are needed to find an “infraction” of the form $y_n.j \sqsupset x_n.j$.

We have to show that the conditions imposed in **11** on x_{n+1} are fulfilled, which, given the definition of y_n, are:

$$y_n \sqsupseteq x_{n+1} \sqsupseteq x_n$$

$$y_n = x_n \Leftarrow x_{n+1} = x_n$$

Proof. Various parts of the proof proceed by case analysis. In the scope of a “Case A” clause, j is the index of some infraction $y_n.j \sqsupset x_n.j$.

First we prove an auxiliary lemma, namely

$$y_n \sqsupseteq x_{n+1} \;\Leftarrow\; y_n \sqsupseteq x_n$$

(Case A)

$$
\begin{array}{ll}
 & y_n \sqsupseteq x_{n+1} \\
\equiv & \qquad \{\text{definition of } x_{n+1} \text{ (Case A)}\} \\
 & y_n \sqsupseteq x_n[.j \mapsto y_n.j] \\
\equiv & \qquad \{\text{definition of } \sqsupseteq_\Pi\} \\
 & \forall(i : i \in I : y_n.i \sqsupseteq x_n[.j \mapsto y_n.j].i) \\
\equiv & \qquad \{\text{range split, 1-pt rule}\} \\
 & \forall(i : i \in I \wedge i \neq j : y_n.i \sqsupseteq x_n[.j \mapsto y_n.j].i) \wedge \\
 & \qquad\qquad y_n.j \sqsupseteq x_n[.j \mapsto y_n.j].j \\
\equiv & \qquad \{\text{definition of } \_[.\_ \mapsto \_]\} \\
 & \forall(i : i \in I \wedge i \neq j : y_n.i \sqsupseteq x_n.i) \;\wedge\; y_n.j \sqsupseteq y_n.j \\
\equiv & \qquad \{\sqsupseteq \text{ is reflexive}\} \\
 & \forall(i : i \in I \wedge i \neq j : y_n.i \sqsupseteq x_n.i) \\
\Leftarrow & \qquad \{\text{constriction}\} \\
 & \forall(i : i \in I : y_n.i \sqsupseteq x_n.i) \\
\equiv & \qquad \{\text{definition of } \sqsupseteq_\Pi\} \\
 & y_n \sqsupseteq x_n
\end{array}
$$

(Case B)

$$
\begin{array}{ll}
 & y_n \sqsupseteq x_{n+1} \\
\equiv & \qquad \{\text{definition of } x_{n+1} \text{ (Case B)}\} \\
 & y_n \sqsupseteq x_n
\end{array}
$$

Now we deal with the components of the conditions on x_{n+1}.

For part "$x_{n+1} \sqsupseteq x_n$" the proof proceeds by case analysis.

(Case A)

$$
\begin{array}{ll}
& x_{n+1} \sqsupseteq x_n \\
\equiv & \quad \{\text{definition of } x_{n+1} \text{ (Case A)}\} \\
& x_n[.j {\mapsto} y_n.j] \sqsupseteq x_n \\
\equiv & \quad \{\text{definition of } \sqsupseteq_\Pi\} \\
& \forall(i : i{\in}I : x_n[.j {\mapsto} y_n.j].i \sqsupseteq x_n.i) \\
\equiv & \quad \{\text{range split, 1-pt rule}\} \\
& \forall(i : i{\in}I \wedge i \neq j : x_n[.j {\mapsto} y_n.j].i \sqsupseteq x_n.i) \wedge x_n[.j {\mapsto} y_n.j].j \sqsupseteq x_n.j \\
\equiv & \quad \{\text{definition of } \_[.\_ {\mapsto} \_]\} \\
& \forall(i : i{\in}I \wedge i \neq j : x_n.i \sqsupseteq x_n.i) \wedge y_n.j \sqsupseteq x_n.j \\
\equiv & \quad \{\sqsupseteq \text{ is reflexive}\} \\
& y_n.j \sqsupseteq x_n.j \\
\Leftarrow & \quad \{\text{definition of } \sqsupset\} \\
& y_n.j \sqsupset x_n.j \\
\equiv & \quad \{\text{Case A}\} \\
& \text{true}
\end{array}
$$

(Case B)

$$
\begin{array}{ll}
& x_{n+1} \sqsupseteq x_n \\
\equiv & \quad \{\text{definition of } x_{n+1} \text{ (Case B)}\} \\
& x_n \sqsupseteq x_n \\
\equiv & \quad \{\sqsupseteq \text{ is reflexive}\} \\
& \text{true}
\end{array}
$$

For part "$y_n \sqsupseteq x_{n+1}$" the proof proceeds by induction.

(Basis)

$$
\begin{array}{ll}
& y_0 \sqsupseteq x_1 \\
\Leftarrow & \quad \{\text{auxiliary lemma}\} \\
& y_0 \sqsupseteq x_0 \\
\equiv & \quad \{\text{definition of } x_0\} \\
& y_0 \perp\!\!\!\perp \sqsupseteq \perp\!\!\!\perp \\
\equiv & \quad \{\perp\!\!\!\perp\text{-characterization}\}
\end{array}
$$

true

(Step)

$$y_{n+1} \sqsupseteq x_{n+2}$$
$\Leftarrow$ {auxiliary lemma}
$$y_{n+1} \sqsupseteq x_{n+1}$$
$\Leftarrow$ {$\sqsupseteq$ is transitive}
$$y_{n+1} \sqsupseteq y_n \wedge y_n \sqsupseteq x_{n+1}$$
$\equiv$ {definition of y_n}
$$F.x_{n+1} \sqsupseteq F.x_n \wedge y_n \sqsupseteq x_{n+1}$$
$\equiv$ {$x_{n+1} \sqsupseteq x_n$ (proved above), F is monotonic}
$$y_n \sqsupseteq x_{n+1}$$

Remark. Since we now have proved both $y_n \sqsupseteq x_{n+1}$ and $x_{n+1} \sqsupseteq x_n$, by the transitivity of $\sqsupseteq$ we also have $y_n \sqsupseteq x_n$.

For part "$y_n = x_n \Leftarrow x_{n+1} = x_n$" the proof proceeds again by case analysis.

(Case A)

$$y_n = x_n$$
$\Leftarrow$ {propositional calculus}
false
$\equiv$ {definition of $\sqsupset$}
$$y_n.j = x_n.j \wedge y_n.j \sqsupset x_n.j$$
$\equiv$ {Case A}
$$y_n.j = x_n.j$$
$\equiv$ {definition of $\_[.\_\mapsto\_]$}
$$x_n[.j \mapsto y_n.j].j = x_n.j$$
$\equiv$ {definition of x_{n+1} (Case A)}
$$x_{n+1}.j = x_n.j$$
$\Leftarrow$ {Leibniz}
$$x_{n+1} = x_n$$

(Case B)

$$
\begin{array}{ll}
 & y_n = x_n \\
\equiv & \quad \{\text{definition of } \sqsupset \} \\
 & y_n \sqsupseteq x_n \ \wedge \ y_n \not\sqsupset x_n \\
\equiv & \quad \{\text{remark above}\} \\
 & y_n \not\sqsupset x_n \\
\Leftarrow & \quad \{\text{by contraposition of } \mathbf{15}\} \\
 & \neg\exists(j : j{\in}I : y_n.j \ \sqsupset \ x_n.j) \\
\equiv & \quad \{\text{Case B}\} \\
 & \text{true}
\end{array}
$$

End of proof.