Least and Greatest Fixpoints in Second-Order Logic

Giuseppe De Giacomo

Construction principle

To define a set Z, here denoted by a predicate $Z(\vec{x})$, we need to say what its elements are.

Construction Principle

The *construction principle* tells us how to obtain these elements recursively.

$$\forall \vec{x}. Z(\vec{x}) \equiv \Phi(Z, \vec{x}) \tag{1}$$

In this case Φ is called a *constructor* for Z.

Any solution of this recursive equation is called a *fixpoint* of the operator Φ.

To define a set Z, here denoted by a predicate $Z(\vec{x})$, we need to say what its elements are, that is we need to choose a solution.

Example: natural numbers

Example (Natural Numbers)

We often define Natural Number as:

- $0 \in \mathbb{N}$
- $\forall x.x \in \mathbb{N} \supset succ(x) \in \mathbb{N}$
- nothing else in $\mathbb{N}$

If we want to fully formalize this definition, we can start by defining the corresponding *construction principle* $\forall \vec{x}.N(\vec{x}) \equiv \Phi(N, \vec{x})$, which in this case is:

$$\forall x.(N(x) \equiv x = 0 \vee \exists y.x = succ(y) \wedge N(y))$$

The *constructor*, $\Phi(N, \vec{x})$, is

$$x = 0 \vee \exists y.x = succ(y) \wedge N(y)$$

Formally $\mathbb{N}(x)$ is the least fixpoint of such constructor, i.e., the smallest solution that satisfy this construction principle.

Least fixpoint

The Tarski-Knaster Theorem guarantees that if the operator Φ is monotone, the equation (1) has both a least and a greatest solution.

A sufficient condition for monotonicity is that all occurrences of Z occur within a even number of negations. This condition must be always satisfied.

Least Fixpoint in Second-Order Logic

Specifically the smallest solution of the equation (1), i.e., the *least fixpoint* is characterized in Second-Order Logic (SOL) as:

$$
\begin{aligned}
Z^{\mu}(\vec{x}) \equiv \ & \forall P.(\\
& \qquad \forall \vec{y}.(\Phi(P, \vec{y}) \supset P(\vec{y})) \\
& \supset P(\vec{x}))
\end{aligned}
\tag{2}
$$

NB: (2) is indeed a re-writing in SOL of Tarski-Knaster's characterization of the least fixpoint:

$$Z^{\mu} = \bigcap \{\mathcal{P} \mid \ \Phi(\mathcal{P}) \subseteq \mathcal{P}\}$$

Example: natural numbers

Example (Natural Numbers)

Formally $\mathbb{N}(x)$ is the least fixpoint of such constructor, i.e., the smallest solution that satisfy the construction principle.
Namely considering that the constructor $\Phi(N, \vec{x})$ is

$$x = 0 \lor \exists y.x = succ(y) \land N(y)$$

We can define Natural Numbers $\mathbb{N}(x)$ in Second Order Logic as:

$$\begin{aligned} \mathbb{N}(x) \equiv \ & \forall N.(\\ & \qquad \forall x.((x = 0 \lor \exists y.x = succ(y) \land N(y)) \supset N(x)) \\ & \supset N(x)) \end{aligned} \tag{3}$$

Note N in the constructor above is not in the scope of any negation, hence the constructor is a (syntactically) monotonic operator, and Tarski-Knaster fixpoint theorem applies: a unique least fixpoint exists.

Induction principle

Let Z^μ be the smallest solution to (1). Then the *induction principle* holds.

Induction Principle

$$\begin{aligned}&\forall P, \vec{x}.(\\ &\qquad \forall \vec{y}.(\Phi(P, \vec{y}) \supset P(\vec{y})) \\ &\supset (Z^\mu(\vec{x}) \supset P(\vec{x})))\end{aligned} \tag{4}$$

i.e., whatever P satisfying the the recursive specification, we have that Z^μ is included in it.

A set Z satisfying construction principle (1) and induction principle (4) is indeed the smallest solution Z^μ to (1), i.e., *least fixpoint* of an operator $\Phi(P, \vec{y})$.

Induction principle: least fixpoint

We can rewrite the induction principle (4) in the following way

$$\begin{aligned}\forall \vec{x}.(Z^\mu(\vec{x}) \supset \ & \forall P.(\\ & \quad \forall \vec{y}.(\Phi(P,\vec{y}) \supset P(\vec{y})) \\ & \supset P(\vec{x})))\end{aligned} \tag{5}$$

Notice that implication in the opposite direction follows from the construction principle (1).

We obtain

$$\begin{aligned}Z^\mu(\vec{x}) \equiv \ & \forall P.(\\ & \quad \forall \vec{y}.(\Phi(P,\vec{y}) \supset P(\vec{y})) \\ & \supset P(\vec{x}))\end{aligned} \tag{6}$$

which is formal definition of a least fixpoint!

Example: natural numbers

Example (Natural Numbers)

The induction principe for the Natural Numbers is:

$$\begin{aligned}
&\forall P.(\\
&\qquad \forall x.((x = 0 \lor \exists y.x = succ(y) \land P(y)) \supset P(x)) \\
&\supset (\mathbb{N}(x) \supset P(x)))
\end{aligned} \tag{7}$$

This formula can be rewritten in a more familia format:

$$\begin{aligned}
&\forall P.(\\
&\qquad \forall x.(\\
&\qquad\qquad (x = 0 \supset P(x)) && \textit{base case} \\
&\qquad\qquad\quad \land \\
&\qquad\qquad (\forall y.x = succ(y) \land P(y) \supset P(x)) && \textit{inductive case} \\
&\supset (\mathbb{N}(x) \supset P(x)))
\end{aligned} \tag{8}$$

This is the formula that we implicitly use when we do prove by induction.

Greatest fixpoint

Consider again the construction principle:

$$\forall \vec{x}. Z(\vec{x}) \equiv \Phi(Z, \vec{x}) \tag{9}$$

Greatest Fixpoint in Second-Order Logic

The greatest solution of the equation (1), i.e., the greatest fixpoint is characterized in Second-Order Logic (SOL) as:

$$\begin{aligned} Z^{\nu}(\vec{x}) \equiv \;& \exists P.(\\ & \qquad \forall \vec{y}. P(\vec{y}) \supset \Phi(P, \vec{y})) \\ & \wedge P(\vec{x}) \end{aligned} \tag{10}$$

NB: (10) is indeed a re-writing in SOL of Tarski-Knaster's characterization of the greatest fixpoint:

$$Z^{\nu} = \bigcup \{\mathcal{P} \mid \mathcal{P} \subseteq \Phi(\mathcal{P})\}$$

Coinduction principle: greatest fixpoint

Let Z^ν is the largest solution of (1). Then the *coinduction principle* holds.

Coinduction Principle

$$\begin{aligned} &\forall P, \vec{x}.(\\ &\qquad \forall \vec{y}.(P(\vec{y}) \supset \Phi(P, \vec{y})) \\ &\supset (P(\vec{x}) \supset Z^\nu(\vec{x}))) \end{aligned} \tag{11}$$

i.e., whatever solution P of the recursive specification we take, Z^ν includes it.

Note that, we can rewrite the coinduction principle (11) in the following way

$$\begin{aligned} &\forall \vec{x}.(\\ &\qquad (\exists P.(\forall \vec{y}.P(\vec{y}) \supset \Phi(P, \vec{y})) \land P(\vec{x})) \\ &\supset Z^\nu(\vec{x})) \end{aligned} \tag{12}$$

Notice that implication in the opposite direction follows from the construction principle (1). Combining the two, we get back the definition of greatest fixpoint 10.

Example: bisimulation

Example (Bisimulation)

We define bisimulation as:

- $$(s,t) \in R \supset \quad \Pi_1(s) = \Pi_2(t) \ \wedge$$
  $$\forall a.\delta_1(s,a,s') \supset \exists t'.\delta_2(t,a,t') \wedge (s',t') \in R \ \wedge$$
  $$\forall a.\delta_2(t,a,t') \supset \exists s'.\delta_1(s,a,s') \wedge (s',t') \in R$$

The corresponding *construction principle*:

$$\begin{aligned} R(s,t) \equiv \quad & (\Pi_1(s) = \Pi_2(t)) \ \wedge \\ & (\forall a.\delta_1(s,a,s') \supset \exists t'.\delta_2(t,a,t') \wedge R(s',t')) \ \wedge \\ & (\forall a.\delta_2(t,a,t') \supset \exists s'.\delta_1(s,a,s') \wedge R(s',t')) \end{aligned}$$

Example: bisimulation

Example (Bisimilarity - Greatest Fixpoint)

Bisimilarity, denoted by $\sim$ is the largest bisimulation, i.e., the largest solution of the construction principle above, i.e., the greatest fixpoint. In second-order logic:

$$
\begin{array}{l}
(\hat{s} \sim \hat{t}) \equiv \\
\qquad \exists R.(\forall s,t.R(s,t) \supset \begin{array}[t]{l} (\Pi_1(s) = \Pi_2(t)) \;\wedge \\ (\forall a.\delta_1(s,a,s') \supset \exists t'.\delta_2(t,a,t') \wedge R(s',t')) \;\wedge \\ (\forall a.\delta_2(t,a,t') \supset \exists s'.\delta_1(s,a,s') \wedge R(s',t'))) \end{array} \\
\qquad\quad \wedge \\
\qquad\quad R(\hat{s},\hat{t})
\end{array}
$$

Example: bisimulation

Example (Coinduction Principle for Bisimilarity)

Bisimilarity, denoted by $\sim$ is the largest bisimulation, i.e., the largest solution of the construction principle above, i.e., the greatest fixpoint. In second-order logic:

$$
\begin{array}{lll}
\forall R, \hat{s}, \hat{t}. & & \\
\quad (\forall s, t. R(s,t) \supset & (\Pi_1(s) = \Pi_2(t)) \;\wedge & \textit{i.e., } R \textit{ is a bisimulation} \\
& (\forall a. \delta_1(s,a,s') \supset \exists t'. \delta_2(t,a,t') \wedge R(s',t')) \;\wedge & \\
& (\forall a. \delta_2(t,a,t') \supset \exists s'. \delta_1(s,a,s') \wedge R(s',t'))) & \\
\supset (R(\hat{s},\hat{t}) \supset (\hat{s} \sim \hat{t})) & &
\end{array}
$$