The Use of the Coinduction Hypothesis in Coinductive Proofs

Anton Setzer

Dept. of Computer Science, Swansea University, Swansea, UK
a.g.setzer@swan.ac.uk

Abstract

For inductively defined sets proofs by induction are carried out using the induction hypothesis. We demonstrate how such a methodology can be applied as well to proofs by coinduction, which can make use of the coinduction hypothesis. We show as well how to apply this methodology to proofs of bisimilarity on transition systems.

The goal of this talk is to transfer the way of carrying out coinductive proofs in type theoretic theorem provers to reasoning in ordinary mathematics. We will use standard set theoretic notion, however use $n : \mathbb{N}$ instead of $n \in \mathbb{N}$.

When reasoning about inductively defined sets we are used to argue informally while referring to the induction hypothesis. When for instance showing $\forall x, y, z : \mathbb{N}.(x + y) + z = x + (y + z)$, we do not first define a set $R := \{z : \mathbb{N} \mid \forall x, y.(x + y) + z = x + (y + z)\}$ and then argue that R is closed under 0 and successor. Instead we show $(x + y) + 0 = x + (y + 0)$ and prove $(x + y) + \mathrm{S}(z) = x + (y + \mathrm{S}(z))$ by using the induction hypothesis $(x + y) + z = x + (y + z)$. Both forms of reasoning are equivalent, but the latter is more light weight and easier to use.

When proving properties about coinductively defined sets, i.e. final coalgebras, we are currently usually following principles which are similar to using for natural numbers the set R being closed under $0, \mathrm{S}$ as above. For instance when showing that two elements of a labelled transition system are bisimilar, one defines a relation on pairs of states of the transition system and shows that it is a bisimulation relation. This makes coinductive reasoning difficult to use.

In [2] we have introduced very general schemata for reasoning corecursively and coinductively by using a coinduction hypothesis in a similar way as one reasons primitive recursively and inductively. The theory was developed for the coinductive version of the Petersson Synek Trees, which capture using containerisation a large class of strictly positive indexed coinductively defined sets. In this talk we will present some examples of how to apply this to more simple examples. The use of the coinduction hypothesis is simplified by making use of the definition of coinductively defined sets as given by their elimination rules [1].

Consider the example of the set of increasing streams $\mathrm{IncStream}(n)$, which is the coinductively defined set indexed over $\mathbb{N}$ given by the eliminators (observations)

$$\begin{array}{lcl} \mathrm{head} & : & (n : \mathbb{N}) \to \mathrm{IncStream}(n) \to \mathbb{N}_{\geq n} \\ \mathrm{tail} & : & (n : \mathbb{N}) \to (s : \mathrm{IncStream}(n)) \to \mathrm{IncStream}(\mathrm{head}(n, s) + 1) \end{array}$$

where $\mathbb{N}_{\geq n} := \{m : \mathbb{N} \mid m \geq n\}$.

The schema for *corecursion* expresses that we can define, assuming a set X and $i : X \to \mathbb{N}$ a function $f : (x : X) \to \mathrm{IncStream}(i(x))$, provided we define for $x : X$

$$\begin{array}{lclcl} \mathrm{head}(i(x), f(x)) & = & m & : & \mathbb{N}_{\geq i(x)} \\ \mathrm{tail}(i(x), f(x)) & = & s & : & \mathrm{IncStream}(m + 1) \end{array}$$

where m and s depend on x, and s can be defined either as an element of $\mathrm{IncStream}(m + 1)$ known before defining f or s can be given by the *corecursion hypothesis* $f(x')$ for some x' s.t. $i(x') = m + 1$.

For instance we can define

$$\begin{array}{lcl}
\mathrm{inc}, \mathrm{inc}', \mathrm{inc}'' & : & (n : \mathbb{N}) \to \mathrm{IncStream}(n) \\
\mathrm{head}(n, \mathrm{inc}(n)) & = & \mathrm{head}(n, \mathrm{inc}'(n)) = \mathrm{head}(n, \mathrm{inc}''(n)) = n \\
\mathrm{tail}(n, \mathrm{inc}(n)) & = & \mathrm{inc}(n+1) \\
\mathrm{tail}(n, \mathrm{inc}'(n)) & = & \mathrm{inc}''(n+1) \\
\mathrm{tail}(n, \mathrm{inc}''(n)) & = & \mathrm{inc}'(n+1)
\end{array}$$

This definition can be made to confirm with the schema by replacing the simultaneous definition of $\mathrm{inc}, \mathrm{inc}', \mathrm{inc}''$ by one function combined : $((n : \mathbb{N}) \times \{\mathrm{inc}, \mathrm{inc}', \mathrm{inc}''\}) \to \mathrm{IncStream}(n)$.

When we have final coalgebras, then we get as well the schema of coinduction. In case of IncStream it is as follows: Assume $i : X \to \mathbb{N}$ and $f, g : (x : X) \to \mathrm{IncStream}(i(x))$. We can prove $\forall x : X.f(x) = g(x)$ by showing the following:

$$\begin{array}{llcl}
\forall x : X. & \mathrm{head}(i(x), f(x)) & = & \mathrm{head}(i(x), g(x)) \\
\forall x : X. & \mathrm{tail}(i(x), f(x)) & = & \mathrm{tail}(i(x), g(x))
\end{array}$$

where for proving $\mathrm{tail}(i(x), f(x)) = \mathrm{tail}(i(x), g(x))$ we can use the fact that $\mathrm{tail}(i(x), f(x)) = f(x')$ and $\mathrm{tail}(i(x), g(x)) = g(x')$ for some x' s.t. $i(x') = \mathrm{head}(i(x), f(x)) + 1$, and by using the *coinduction hypothesis* $f(x') = g(x')$.

For instance we can prove by coinduction on IncStream

$$(\forall n : \mathbb{N}.\mathrm{inc}(n) = \mathrm{inc}'(n)) \land (\forall n : \mathbb{N}.\mathrm{inc}(n) = \mathrm{inc}''(n))$$

as follows:

$$\begin{array}{lclclclcl}
\mathrm{head}(\mathrm{inc}(n)) & = & n & = & \mathrm{head}(\mathrm{inc}'(n)) & = & n & = & \mathrm{head}(\mathrm{inc}''(n)) \\
\mathrm{tail}(\mathrm{inc}(n)) & = & \mathrm{inc}(n+1) & \overset{\mathrm{co-IH}}{=} & \mathrm{inc}''(n+1) & = & \mathrm{tail}(\mathrm{inc}'(n)) & & \\
\mathrm{tail}(\mathrm{inc}(n)) & = & \mathrm{inc}(n+1) & \overset{\mathrm{co-IH}}{=} & \mathrm{inc}'(n+1) & = & \mathrm{tail}(\mathrm{inc}''(n)) & &
\end{array}$$

The above principle applies as well to proofs of bisimilarity $\sim$ for labelled transition systems. Restricting ourselves to the unlabelled case, let a transition system be given by a set T and a relation $\longrightarrow \subseteq \mathrm{T} \times \mathrm{T}$. The schema for proving bisimilarity on $(\mathrm{T}, \longrightarrow)$ is as follows: Assume $f, g : X \to \mathrm{T}$. We can prove $\forall x \in X.f(x) \sim g(x)$ by giving:

- for $f(x) \longrightarrow t$ a t' s.t. $g(x) \longrightarrow t'$ and $t \sim t'$,
- for $g(x) \longrightarrow t'$ a t s.t. $f(x) \longrightarrow t$ and $t \sim t'$.

where in case $t = f(x')$ and $t' = g(x')$ for some x' we are allowed to use for proving $t \sim t'$ the *coinduction-hypothesis* $f(x') \sim g(x')$.

For instance, let $\mathrm{T} := \{*\} \cup \mathbb{N}$, $\longrightarrow$ given by $* \longrightarrow *$, $n \longrightarrow n + 1$.

We show $\forall n \in \mathbb{N}.* \sim n$:

- Assume $* \longrightarrow x$. Then $x = *$. $n \longrightarrow n + 1$ and by co-IH $* \sim n + 1$.
- Assume $n \longrightarrow x$. Then $x = n + 1$. $* \longrightarrow *$ and by co-IH $* \sim n + 1$.

References

[1] A. Abel, B. Pientka, D. Thibodeau, and A. Setzer. Copatterns: Programming infinite structures by observations. In R. Giacobazzi and R. Cousot, editors, *Proceedings of POPL '13*, pages 27–38. ACM, 2013.

[2] A. Setzer. How to reason coinductively informally. To appear in: Reinhard Kahle, Thomas Strahm, and Thomas Studer (Eds.): Advances in Proof Theory, Birkhäuser, 2016.