Slides for the book
“An Introduction to Bisimulation and Coinduction”

Part II: coinduction and the duality with induction

Davide Sangiorgi (University of Bologna/INRIA)

Induction and coinduction

- examples
- duality
- fixed-point theory

Examples of induction and coinduction

Mathematical induction

To prove a property for all natural numbers:

1. Show that **the property holds at 0** (basis)
2. Show that, **whenever the property holds at n, it also holds at $n+1$** (inductive part)

In a variant, step (2) becomes:

Show that, whenever the property holds at all natural less than or equal to n, then it also holds at $n+1$

NB: other variants are possible, modifying for instance the basis

Example of mathematical induction

$$1 + 2 + \ldots + n = \frac{n \times (n+1)}{2}$$

Basis: $1 = \dfrac{1 \times 2}{2}$

Inductive step: (assume true at n, prove statement for $n+1$)

$$\begin{aligned}
1 + 2 + \ldots + n + (n+1) &= \text{(inductive hypothesis)} \\
\frac{n \times (n+1)}{2} + (n+1) &= \\
\frac{n \times (n+1)}{2} + \frac{2 \times (n+1)}{2} &= \\
\frac{n \times (n+1) + 2 \times (n+1)}{2} &= \\
\frac{(n+1) \times (n+2)}{2} &= \frac{(n+1) \times ((n+1)+1)}{2}
\end{aligned}$$

A non-example of mathematical induction

Statement: **All men have eyes of the same color**

Basis: trivial (with only one man, there is only one color)

Inductive step: (assume true at n, prove statement for $n+1$)

Order the $n+1$ men, in the positions $\{1, 2, \ldots, n+1\}$

Consider now the two subsets of men in positions $\{1, 2, \ldots, n\}$ and $\{2, \ldots, n+1\}$

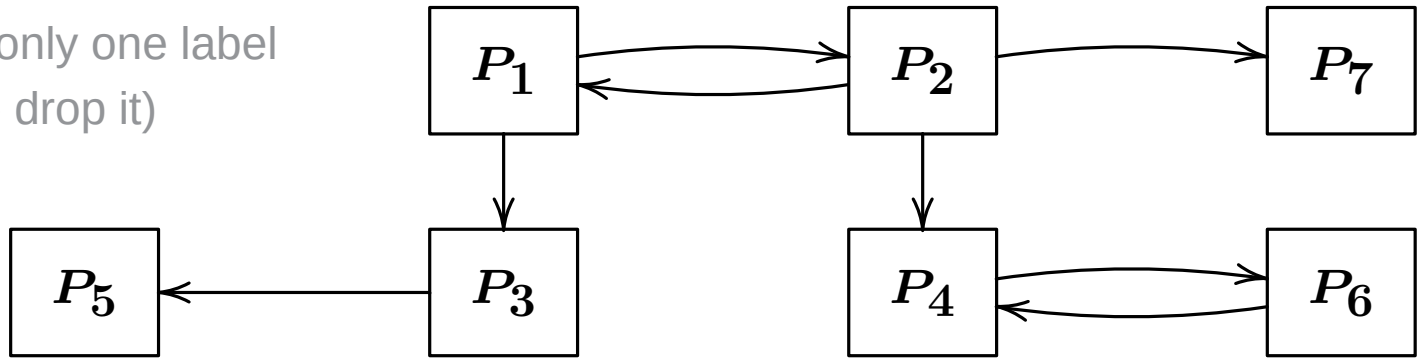
Each subset has n elements.

By the inductive hypothesis, the men in each subset have the eyes of the same color

The two sets overlap, so there is only one color for the eyes of the $n+1$ men.

Rule induction: finite traces (may termination)

(assume only one label
hence we drop it)



A process **stopped**: it cannot do any transitions

P has a **finite trace**, written $P\downarrow$, if P has a finite sequence of transitions that lead to a stopped process

Examples: P_1, P_2, P_3, P_5, P_7 (how many finite traces for P_2?)

(inductive definition of $\downarrow$) $P\downarrow$ if
(1) P is stopped
(2) $\exists\, P'$ with $P \longrightarrow P'$ and $P'\downarrow$

as rules:

$$\frac{P \text{ stopped}}{P\downarrow} \qquad \frac{P \longrightarrow P' \qquad P'\downarrow}{P\downarrow}$$

What is a set inductively defined by a set of rules?

...**later**, using some (very simple) fixed-point and lattice theory

Now: 3 equivalent readings of inductive sets, informally

1. sets of elements with a certain **proof trees**
2. sets satisfying a certain **closure property**
   (from which we derive the familiar method of proofs by induction)
3. sets obtained via a certain **iteration schema**

We will do the same for coinduction

Then we formally justify the 3 readings, from fixed-point theory
First (2), then (3), then (1).

We will also see another reading, as **games**

Equivalent readings for $\downarrow$

$$\frac{P \text{ stopped}}{P \downarrow}\ (\mathsf{AX}) \qquad\qquad \frac{P \longrightarrow P' \qquad P' \downarrow}{P \downarrow}\ (\mathsf{INF})$$

– The processes obtained with a finite proof from the rules

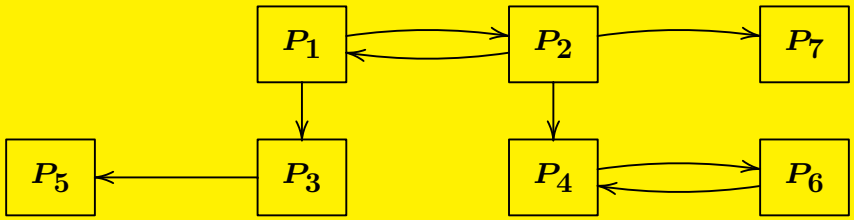
Equivalent readings for $\downarrow$

$$\frac{P \text{ stopped}}{P \downarrow}\ (\mathsf{AX}) \qquad\qquad \frac{P \longrightarrow P' \qquad P' \downarrow}{P \downarrow}\ (\mathsf{INF})$$

– The processes obtained with a finite proof from the rules

Example

$$\frac{P_1 \longrightarrow P_2 \qquad \dfrac{P_2 \longrightarrow P_7 \qquad \dfrac{P_7 \text{ stopped}}{P_7 \downarrow}\ (\mathsf{AX})}{P_2 \downarrow}\ (\mathsf{INF})}{P_1 \downarrow}\ (\mathsf{INF})$$



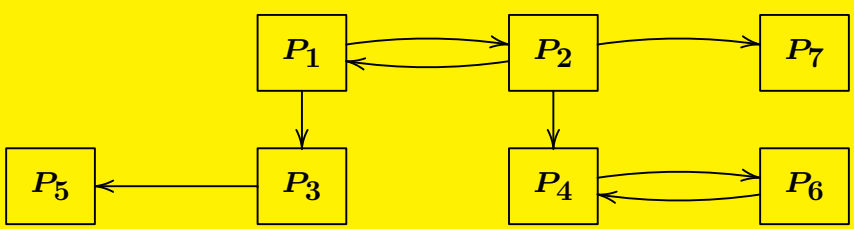
Equivalent readings for $\downarrow$

$$\frac{P \text{ stopped}}{P \downarrow}\ (\mathsf{AX}) \qquad\qquad \frac{P \longrightarrow P' \qquad P' \downarrow}{P \downarrow}\ (\mathsf{INF})$$

– The processes obtained with a finite proof from the rules

Example **(another proof for P_1; how many other proofs?)** :

$$\dfrac{P_1 \longrightarrow P_2 \qquad \dfrac{P_2 \longrightarrow P_1 \qquad \dfrac{P_1 \longrightarrow P_2 \qquad \dfrac{P_2 \longrightarrow P_7 \qquad \dfrac{P_7 \text{ stopped}}{P_7 \downarrow}}{P_2 \downarrow}}{P_1 \downarrow}}{P_2 \downarrow}}{P_1 \downarrow}$$



Equivalent readings for $\downarrow$

$$\frac{P \text{ stopped}}{P \downarrow}\ (\text{AX}) \qquad\qquad \frac{P \longrightarrow P' \qquad P' \downarrow}{P \downarrow}\ (\text{INF})$$

- The processes obtained with a finite proof from the rules
- the **smallest** set of processes that is **closed forward under the rules**; i.e., the smallest subset S of *Pr* (all processes) such that
  - $*$ all stopped processes are in S;
  - $*$ if there is P' with $P \longrightarrow P'$ and $P' \in S$, then also $P \in S$.

Equivalent readings for $\downarrow$

$$\frac{P \text{ stopped}}{P \downarrow}\ (\text{AX}) \qquad\qquad \frac{P \longrightarrow P' \qquad P' \downarrow}{P \downarrow}\ (\text{INF})$$

- The processes obtained with a finite proof from the rules
- the **smallest** set of processes that is **closed forward under the rules**;
  i.e., the smallest subset S of *Pr* (all processes) such that
  - $*$ all stopped processes are in S;
  - $*$ if there is P' with $P \longrightarrow P'$ and $P' \in S$, then also $P \in S$.

Hence a proof technique for $\downarrow$ (rule induction):

given a property T on the processes (a subset of processes),
to prove $\downarrow \subseteq T$ (all processes in $\downarrow$ have the property)
show that T is closed forward under the rules.

Example of rule induction for finite traces

A partial function f, from processes to integers, that satisfies the following conditions:

$$
\begin{array}{rcll}
f(P) & = & 0 & \text{if } P \text{ is stopped} \\
f(P) & = & \mathtt{min}\{f(P') + 1 \mid P \longrightarrow P' \text{ for some } P' & \\
 & & \quad \text{and } f(P') \text{ is defined } \} & \text{otherwise}
\end{array}
$$

(f can have any value, or even be undefined, if the set on which the `min` is taken is empty)

We wish to prove f defined on processes with a finite trace (i.e., $\mathtt{dom}(\downharpoonleft) \subseteq \mathtt{dom}(f)$)

We can show that $\mathtt{dom}(f)$ is closed forward under the rules defining $\downharpoonleft$.

Proof:

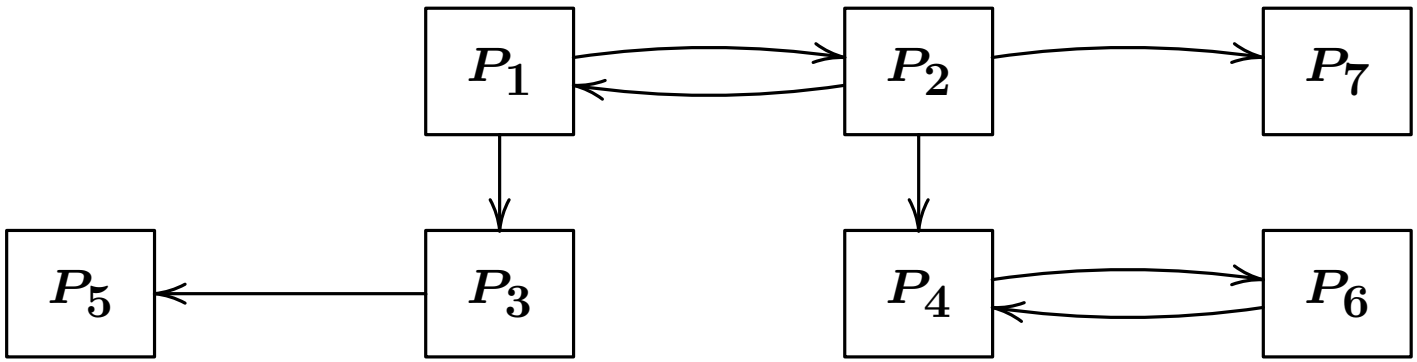
1. $f(P)$ is defined whenever P is stopped;
2. if there is P' with $P \longrightarrow P'$ and $f(P')$ is defined, then also $f(P)$ is defined.

Equivalent readings for $\downarrow$

$$\frac{P \text{ stopped}}{P \downarrow}\ (\mathsf{AX}) \qquad\qquad \frac{P \longrightarrow P' \qquad P' \downarrow}{P \downarrow}\ (\mathsf{INF})$$

- The processes obtained with a finite proof from the rules
- the **smallest** set of processes that is **closed forward under the rules**;
  i.e., the smallest subset S of *Pr* (all processes) such that
  - $*$ all stopped processes are in S;
  - $*$ if there is P' with $P \longrightarrow P'$ and $P' \in S$, then also $P \in S$.
- (iterative construction)
  Start from $\emptyset$;
  add all objects as in the axiom;
  repeat adding objects following the inference rule **forwards**

Rule coinduction definition: ω-traces (non-termination)



P has an **ω-trace**, written $P\uparrow$, if it there is an infinite sequence of transitions starting from P.

Examples: P_1, P_2, P_4, P_6

Coinductive definition of $\uparrow$:

$$\frac{P \longrightarrow P' \qquad P'\uparrow}{P\uparrow}$$

Equivalent readings for $\downarrow$

$$\frac{P \longrightarrow P' \qquad P' \downarrow}{P \downarrow}$$

- The processes obtained with an **infinite** proof from the rules

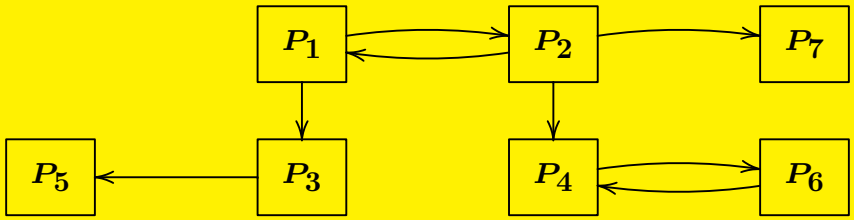
Equivalent readings for $\downarrow$

$$\frac{P \longrightarrow P' \qquad P' \downarrow}{P \downarrow}$$

– The processes obtained with an **infinite** proof from the rules

Example

$$\frac{P_1 \longrightarrow P_2 \qquad \dfrac{P_2 \longrightarrow P_1 \qquad \dfrac{P_1 \longrightarrow P_2 \qquad \dfrac{\vdots}{P_2 \downarrow}}{P_1 \downarrow}}{P_2 \downarrow}}{P_1 \downarrow}$$



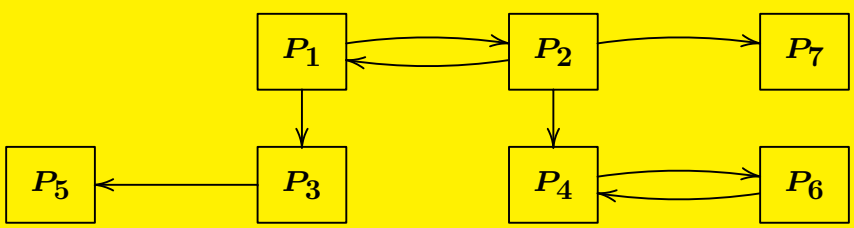
Equivalent readings for $\downarrow$

$$\frac{P \longrightarrow P' \qquad P' \downarrow}{P \downarrow}$$

– The processes obtained with an **infinite** proof from the rules

An invalid proof:

$$\frac{P_1 \longrightarrow P_3 \qquad \dfrac{P_3 \longrightarrow P_5 \qquad \dfrac{??}{P_5 \downarrow}}{P_3 \downarrow}}{P_1 \downarrow}$$



Equivalent readings for $\downarrow$

$$\frac{P \longrightarrow P' \qquad P' \downarrow}{P \downarrow}$$

- The processes obtained with an **infinite** proof from the rules
- the **largest** set of processes that is **closed backward under the rule**; i.e., the largest subset S of processes such that if $P \in S$ then
  $*$ there is P' such that $P \longrightarrow P'$ and $P' \in S$.

Equivalent readings for $\upharpoonright$

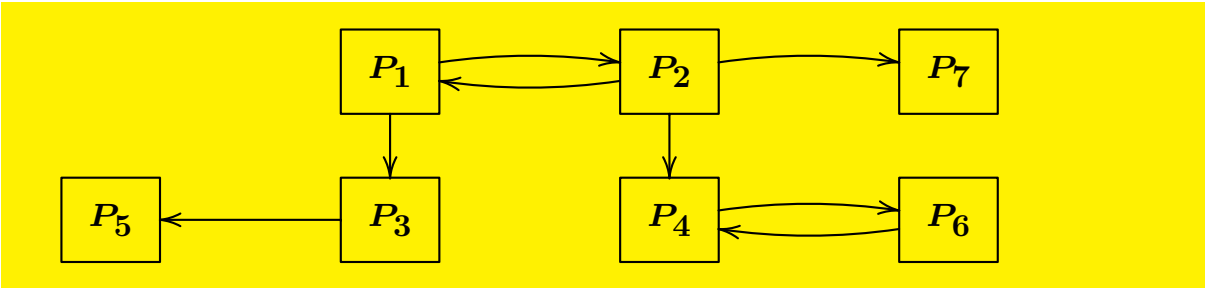
$$\frac{P \longrightarrow P' \qquad P' \upharpoonright}{P \upharpoonright}$$

- The processes obtained with an **infinite** proof from the rules
- the **largest** set of processes that is **closed backward under the rule**; i.e., the largest subset S of processes such that if $P \in S$ then
  $*$ there is P' such that $P \longrightarrow P'$ and $P' \in S$.

Hence a proof technique for $\upharpoonright$ (rule coinduction):

to prove that each process in a set T has an ω-trace
show that T is closed backward under the rule.

Example of rule coinduction for ω-traces



$$\frac{P \longrightarrow P' \qquad P' \restriction}{P \restriction}$$

Suppose we want to prove $P_1 \restriction$

Proof

$T = \{P_1, P_2\}$ is closed backward :

$$\frac{P_1 \longrightarrow P_2 \qquad P_2 \in T}{P_1 \in T} \qquad\qquad \frac{P_2 \longrightarrow P_1 \qquad P_1 \in T}{P_2 \in T}$$

Another choice: $T = \{P_1, P_2, P_4, P_6\}$ (correct, but more work in the proof)

Would $T = \{P_1, P_2, P_4\}$ or $T = \{P_1, P_2, P_3\}$ be correct?

ω-traces in the bisimulation style

A predicate S on processes is **ω-closed** if whenever $P \in S$:
– there is $P' \in S$ such that $P \longrightarrow P'$.
P has an **ω-trace**, written $P \upharpoonright$, if $P \in S$, for some ω-closed predicate S.

The proof technique is explicit

Compare with the definition of bisimilarity:

A relation $\mathcal{R}$ on processes is a **bisimulation** if whenever $P \mathcal{R} Q$:
1. $\forall \mu, P'$ s.t. $P \xrightarrow{\mu} P'$, then $\exists Q'$ such that $Q \xrightarrow{\mu} Q'$ and $P' \mathcal{R} Q'$;
2. $\forall \mu, Q'$ s.t. $Q \xrightarrow{\mu} Q'$, then $\exists P'$ such that $P \xrightarrow{\mu} P'$ and $P' \mathcal{R} Q'$.

P and Q are **bisimilar**, written $P \sim Q$, if $P \mathcal{R} Q$, for some bisimulation $\mathcal{R}$.

Equivalent readings for $\downarrow$

$$\frac{P \longrightarrow P' \qquad P' \downarrow}{P \downarrow}$$

- The processes obtained with an **infinite** proof from the rules
- the **largest** set of processes that is **closed backward under the rule**; i.e., the largest subset S of processes such that if $P \in S$ then
  - there is $P' \in S$ such that $P \longrightarrow P'$.
- (iterative construction) start with the set *Pr* of all processes; repeatedly remove a process P from the set if one of these applies (the **backward closure** fails):
  - P has no transitions
  - all transitions from P lead to derivatives that are not anymore in the set.

An inductive definition: finite lists over a set A

$$\frac{}{\texttt{nil} \in \mathcal{L}} \qquad\qquad \frac{\ell \in \mathcal{L} \qquad a \in A}{\langle a \rangle \bullet \ell \in \mathcal{L}}$$

3 equivalent readings (in the **"forward"** direction):

- The objects obtained with a finite proof from the rules
- The smallest set closed forward under these rules

> A set T is closed forward if:
> - $\texttt{nil} \in T$
> - $\ell \in T$ implies $\langle a \rangle \bullet \ell \in T$, for all $a \in A$

Inductive proof technique for lists: Let T be a predicate (a property) on lists. To prove that T holds on all lists, prove that T is closed forward

- (iterative construction) Start from $\emptyset$; add all objects as in the axiom; repeat adding objects following the inference rule forwards

A coinductive definition: finite and infinite lists over A

$$\frac{}{\texttt{nil} \in \mathcal{L}} \qquad\qquad \frac{\ell \in \mathcal{L} \qquad a \in A}{\langle a \rangle \bullet \ell \in \mathcal{L}}$$

3 equivalent readings (in the **"backward"** direction) :

- The objects that are conclusion of a finite or infinite proof from the rules
- The largest set closed backward under these rules

> A set T is closed backward if $\forall t \in T$:
> – either $t = \texttt{nil}$
> – or $t = \langle a \rangle \bullet \ell$, for some $\ell \in T$ and $a \in A$

Coinduction proof method: to prove that ℓ is a finite or infinite list, find a set D with $\ell \in D$ and D closed backward

- $X =$ all (finite and infinite) strings of $A \cup \{\texttt{nil}, \langle, \rangle, \bullet\}$
  Start from X (all strings) and keep removing strings, following the backward-closure

An inductive definition: convergence, in λ-calculus

Set of λ-terms (an inductive def!) $\quad e ::= x \mid \lambda x.\, e \mid e_1(e_2)$

Convergence to a value ($\Downarrow$), on closed λ-terms, call-by-name:

$$\frac{}{\lambda x.\, e \Downarrow \lambda x.\, e} \qquad \frac{e_1 \Downarrow \lambda x.\, e_0 \qquad e_0\{^{e_2}/_{x}\} \Downarrow e'}{e_1(e_2) \Downarrow e'}$$

As before, $\Downarrow$ can be read in terms of finite proofs, limit of an iterative construction, or smallest set closed forward under these rules

$\Downarrow$ is the smallest relation $\mathcal{S}$ on (closed) λ-terms s.t.

– $\lambda x.\, e \;\mathcal{S}\; \lambda x.\, e$ for all abstractions,

– if $e_1 \;\mathcal{S}\; \lambda x.\, e_0$ and $e_0\{^{e_2}/_{x}\} \;\mathcal{S}\; e'$ then also $e_1(e_2) \;\mathcal{S}\; e'$.

A coinductive definition: divergence in the λ-calculus

Divergence ($\Uparrow$), on closed λ-terms, call-by-name:

$$\frac{e_1 \Uparrow}{e_1(e_2) \Uparrow} \qquad \frac{e_1 \Downarrow \lambda x.\, e_0 \qquad e_0\{^{e_2}/_{x}\} \Uparrow}{e_1(e_2) \Uparrow}$$

The 'closed backward' reading:

> $\Uparrow$ is the *largest* predicate on λ-terms that is closed backward under these rules; i.e., the largest subset D of λ-terms s.t. if $e \in D$ then
> – either $e = e_1(e_2)$ and $e_1 \in D$,
> – or $e = e_1(e_2)$, $e_1 \Downarrow \lambda x.\, e_0$ and $e_0\{^{e_2}/_{x}\} \in D$.

Coinduction proof technique :

to prove $e \Uparrow$, find $E \subseteq \Lambda$ closed backward and with $e \in E$

What is the smallest predicate closed backward?

The duality induction/coinduction

Constructors/destructors

- An inductive definition tells us what are the **constructors** for generating all the elements (cf: the forward closure).
- A coinductive definition tells us what are the **destructors** for decomposing the elements (cf: the backward closure).

> The destructors show what we can **observe** of the elements
> (think of the elements as black boxes;
> the destructors tell us what we can do with them;
> this is clear in the case of infinite lists).

Definitions given by means of rules

- if the definition is **inductive**, we look for the **smallest** universe in which such rules live.
- if it is **coinductive**, we look for the **largest** universe.
- the **inductive proof principle** allows us to infer that the **inductive set is included in a set** (ie, has a given property) by proving that the set satisfies the **forward closure**;
- the **coinductive proof principle** allows us to infer that **a set is included in the coinductive set** by proving that the given set satisfies the **backward closure**.

Forward and backward closures

A set T being closed forward intuitively means that

> *for each* **rule whose premise is satisfied in T**
> *there is* **an element of T**
> *such that the element is the conclusion of the rule.*

In the backward closure for T, the order between the two quantified entities is swapped:

> *for each* **element of T**
> *there is* **a rule whose premise is satisfied in T**
> *such that the element is the conclusion of the rule.*

In fixed-point theory, the duality between forward and backward closure will the duality between pre-fixed points and post-fixed points.

Congruences vs bisimulation equivalences

Congruence: an equivalence relation
that respects the constructors of a language

Example (λ-calculus)

Consider the following rules, acting on pairs of (open) λ-terms:

$$\frac{}{(x, x)} \qquad \frac{(e_1, e_2)}{(e\ e_1, e\ e_2)} \qquad \frac{(e_1, e_2)}{(e_1\ e, e_2\ e)} \qquad \frac{(e_1, e_2)}{(\lambda x.\, e_1, \lambda x.\, e_2)}$$

A congruence: an equivalence relation closed forward under the rules

The smallest such relation is **syntactic equality**: the **identity relation**

In other words, congruence rules express syntactic constraints

Bisimulation equivalence: an equivalence relation that respects the destructors

Example (λ-calculus, call-by-name)

Consider the following rules

$$\frac{e_1 \Uparrow \qquad e_2 \Uparrow}{(e_1, e_2)} \quad e_1, e_2 \text{ closed}$$

$$\frac{e_1 \Downarrow \lambda x.\, e_1' \qquad e_2 \Downarrow \lambda x.\, e_2' \qquad \cup_{e''} \{(e_1'\{e''/x\}, e_2'\{e''/x\})\}}{(e_1, e_2)} \quad e_1, e_2, e'' \text{ closed}$$

$$\frac{\cup_{\sigma} \{(e_1\sigma, e_2\sigma)\}}{(e_1, e_2)} \quad e_1, e_2 \text{ non closed, } \sigma \text{ closing substitution for } e_1, e_2$$

A bisimulation equivalence: an equivalence relation closed backward under the rules

The largest such relation is **semantic equality**: **bisimilarity**

In other words, the bisimulation rules express semantic constraints

Substitutive relations vs bisimulations

In the duality between congruences and bisimulation equivalences, the equivalence requirement is not necessary.

Leave it aside, we obtaining the duality between **bisimulations** and **substitutive relations**

> a relation is substitutive if whenever s and t are related,
> then any term t' must be related to a term s'
> obtained from t' by replacing occurrences of t with s

Bisimilarity is a congruence

To be useful, a bisimilarity on a term language should be a congruence

This leads to proofs where inductive and coinductive techniques are intertwined

In certain languages, for instance higher-order languages, such proofs may be hard, and how to best combine induction and coinduction remains a research topic.

What makes the combination delicate is that the rules on which congruence and bisimulation are defined — the rules for syntactic and semantic equality — are different.

Summary of the dualities

inductive definition	coinductive definition
induction proof principle	coinduction proof principle
constructors	observations
smallest universe	largest universe
’forward closure’ in rules	’backward closure’ in rules
congruence	bisimulation equivalence
substitutive relation	bisimulation
identity	bisimilarity
least fixed point	greatest fixed point
pre-fixed point	post-fixed point
algebra	coalgebra
syntax	semantics
semi-decidable set	cosemi-decidable set
strengthening of the candidate in proofs	weakening of the candidate in proofs

We have seen:

- examples of induction and coinduction
- 3 readings for the sets inductively and coinductively obtained from a set of rules
- justifications for the induction and coinduction proof principles
- the duality between induction and coinduction, informally

Remaining questions

- **What is the definition of an inductive set?**
- **From this definition, how do we derive the previous 3 readings for sets inductively and coinductively obtained from a set of rules?**
- **How is the duality induction/coinduction formalised?**

What follows answers these questions. It is a simple application of fixed-point theory on complete lattices.

To make things simpler, we work on **powersets** and **fixed-point theory**.
(It is possible to be more general, working with universal algebras or category theory.)

Complete lattices and fixed-points

Complete lattices

The important example of complete lattice for us: **powersets**.

For a given set X, the powerset of X, written $\wp(X)$, is

$$\wp(X) \stackrel{\text{def}}{=} \{T \mid T \subseteq X\}$$

$\wp(X)$ is a complete lattice because:

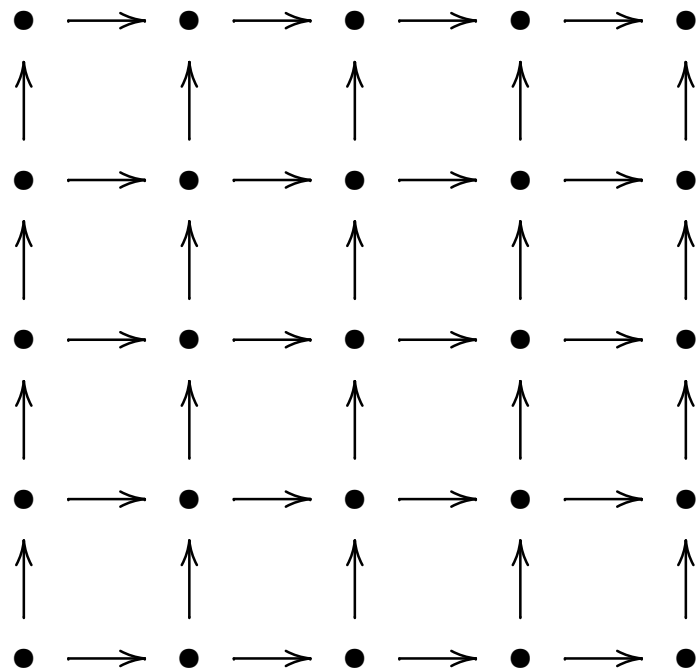
- it comes with a relation $\subseteq$ (set inclusion) that is reflexive, transitive, and antisymmetric.
- it is closed under union and intersection

($\cup$ and $\cap$ give least upper bounds and greatest lower bounds for $\subseteq$)

A **partially ordered set** (or **poset**): a non-empty set with a relation on its elements that is reflexive, transitive, and antisymmetric.

A **complete lattice**: a poset with all joins (least upper bounds) and (hence) also all meets (greatest lower bounds).

Example of a complete lattice



Two points x, y are in the relation $\leq$ if there is a path from x to y following the directional edges

(a path may also be empty, hence $x \leq x$ holds for all x)

A partially ordered set that is not a complete lattice

$$\begin{tikzcd}
a & b \\
c \arrow[u] \arrow[ur] & d \arrow[u] \arrow[l] \\
e \arrow[u] & f \arrow[u]
\end{tikzcd}$$

Again, $x \leq y$ if there is a path from x to y

Bounds, meets, and joins

L poset, $S \subseteq L$: then $y \in L$ is an **upper bound of S** if $x \leq y \; \forall \; x \in S$.

The dual: a **lower bound of S**: (a $y \in L$ with $y \leq x$ for all $x \in S$)

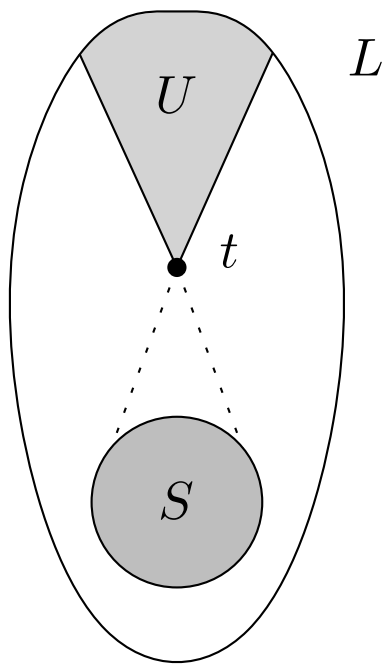
The **least upper bound** (also called the **join**) of S: an upper bound y with $y \leq z \; \forall$ upper bounds z of S

The dual: of these concepts gives us the **greatest lower bound** (or **meet**)

Example:

U = the upper bounds for S

t = the least upper bound



The Fixed-point Theorem

NB: Complete lattices are "dualisable" structures: reverse the arrows and you get another complete lattice. Similarly, statements on complete lattices can be dualised.

For simplicity, we will focus on complete lattices produced by the powerset construction. But all statements can be generalised to arbitrary complete lattices

Given a function F on a complete lattice:

- F is **monotone** if $x \leq y$ implies $F(x) \leq F(y)$, for all x, y.
- x is a **pre-fixed point** of F if $F(x) \leq x$.
  Dually, x is a **post-fixed point** if $x \leq F(x)$.
- x is a **fixed point** of F if $F(x) = x$ (it is both pre- and post-fixed point)
- The set of fixed points of F may have a least element, the **least fixed point**, and a greatest element, the **greatest fixed point**

Example

the poset of the (positive) natural numbers with $n \leq m$ if n divides m

For $S = \{4, 8, 16\}$:

- $\{1, 2, 4\}$ = the set of lower bounds of S

- 4 is the least element in S and its meet;

- 16 is the greatest element and the join

For $S = \{2, 3, 4\}$:

- 1 is the only lower bound and the meet;

- there is no least element; any multiple of 12 is an upper bound, 12 is the join; no greatest element

The endofunction F where $F(n)$ is the sum of the factors of n that are different from n, with the exception of 1 that is mapped onto itself

eg: $F(1) = 1, F(2) = 1, F(3) = 1, F(4) = 3, F(6) = 6$.

Then $1, 2, 3, 6$ are pre-fixed points, and $1, 6$ fixed points.

Exercise

- Is the set of all natural numbers a complete lattice?
- Is it a lattice (that is, is a poset in which all *pairs* of elements have a join)?
- Can we add elements to the set of natural numbers so as to make it a complete lattice?

Exercise

1. Show that if F is a monotone endofunction on a complete lattice, and x and y are post-fixed points of F, then also $\cup\{x, y\}$ is a post-fixed point.
2. Generalise the previous point to an arbitrary set S of post-fixed points: $\cup S$ is also a post-fixed point. Then dualise the result to pre-fixed points.

For simplicity, we discuss the theorem on the complete lattices generated by the powerset construction

Theorem [Fixed-point Theorem] If $F : \wp(X) \to \wp(X)$ is monotone, then

$$\mathtt{lfp}(F) = \bigcap\{T \mid F(T) \subseteq T\}$$

$$\mathtt{gfp}(F) = \bigcup\{T \mid T \subseteq F(T)\}$$

(the meet of the pre-fixed points, the join of the post-fixed points)

NB: the theorem actually says more: the set of fixed points is itself a complete lattice, and the same for the sets of pre-fixed points and post-fixed points.

Proof of the Fixed-point Theorem

We consider one part of the statement (the other part is dual), namely

$$\mathtt{gfp}(F) = \bigcup\{S \mid S \subseteq F(S)\}$$

Set $T = \bigcup\{S \mid S \subseteq F(S)\}$. We have to show T fixed point (it is then the greatest: any other fixed point is a post-fixed point, hence contained in T)

Proof of $T \subseteq F(T)$

For each S s.t. $S \subseteq F(S)$ we have:

	$S \subseteq T$	(def of T as a union)
hence	$F(S) \subseteq F(T)$	(monotonicity of F)
hence	$S \subseteq F(T)$	(since S is a post-fixed point)

We conclude $F(T) \supseteq \bigcup\{S \mid S \subseteq F(S)\} = T$

Proof of the Fixed-point Theorem

We consider one part of the statement (the other part is dual), namely

$$\mathtt{gfp}(F) = \bigcup\{S \mid S \subseteq F(S)\}$$

Set $T = \bigcup\{S \mid S \subseteq F(S)\}$. We have to show T fixed point (it is then the greatest: any other fixed point is a post-fixed point, hence contained in T)

Proof of $F(T) \subseteq T$

We have	$T \subseteq F(T)$	(just proved)
hence	$F(T) \subseteq F(F(T))$	(monotonicity of F)
that is,	$F(T)$ is a post-fixed point	

Done, by definition of T as a union of the post-fixed points.

Sets coinductively and inductively defined by F

Definition Given a complete lattice produced by the powerset construction, and an endofunction F on it, the sets:

$$F_{\mathtt{ind}} \stackrel{\text{def}}{=} \bigcap \{x \mid F(x) \subseteq x\}$$

$$F_{\mathtt{coind}} \stackrel{\text{def}}{=} \bigcup \{x \mid x \subseteq F(x)\}$$

are the sets **inductively defined by F**, and **coinductively defined by F**.

By the Fixed-point Theorem, when F monotone:

$$\begin{aligned} F_{\mathtt{ind}} &= \mathtt{lfp}(F) \\ &= \text{least pre-fixed point of } F \end{aligned}$$

$$\begin{aligned} F_{\mathtt{coind}} &= \mathtt{gfp}(F) \\ &= \text{greatest post-fixed point of } F \end{aligned}$$

Next objective

We wish to derive the reading (2) of inductive sets
(the smallest sets ‘close forward’ wrt some rules)

And dually for coinductive sets

We have to show:

a set of rules	$\Leftrightarrow$	a monotone function on a complete lattice
a forward closure for the rules	$\Leftrightarrow$	a pre-fixed point for the function
a backward closure for the rules	$\Leftrightarrow$	a post-fixed point for the function

NB: all inductive and coinductive definitions can be given in terms of rules

Definitions by means of rules

Given a set X, a **ground rule on X** is a pair (S, x) with $S \subseteq X$ and $x \in X$

We can write a rule (S, x) as

$$\frac{x_1 \ldots x_n \ldots}{x} \qquad \text{where } \{x_1, \ldots, x_n, \ldots\} = S.$$

A rule $(\emptyset, x)$ is an **axiom**

Definitions by means of rules

Given a set X, a **ground rule on X** is a pair (S, x) with $S \subseteq X$ and $x \in X$

We can write a rule (S, x) as

$$\frac{x_1 \ldots x_n \ldots}{x} \qquad \text{where } \{x_1, \ldots, x_n, \ldots\} = S.$$

A rule $(\emptyset, x)$ is an **axiom**

NB: previous rules, eg $\dfrac{P \longrightarrow P' \qquad P' \uparrow}{P \uparrow}$ were not ground (P, P' are metavariables)

The translation to ground rules is trivial (take all valid instantiations)

Definitions by means of rules

Given a set X, a **ground rule on X** is a pair (S, x) with $S \subseteq X$ and $x \in X$

We can write a rule (S, x) as

$$\frac{x_1 \ldots x_n \ldots}{x} \qquad \text{where } \{x_1, \ldots, x_n, \ldots\} = S.$$

A rule $(\emptyset, x)$ is an **axiom**

A set $\mathcal{R}$ of rules on X yields a monotone endofunction $\Phi_{\mathcal{R}}$, called the **functional of $\mathcal{R}$** (or **rule functional**), on the complete lattice $\wp(X)$, where

$$\Phi_{\mathcal{R}}(T) = \{x \mid (T', x) \in \mathcal{R} \text{ for some } T' \subseteq T\}$$

Exercise Show $\Phi_{\mathcal{R}}$ monotone, and that every monotone operator on $\wp(X)$ can be expressed as the functional of some set of rules.

By the Fixed-point Theorem there are least fixed point and greatest fixed point, $\mathtt{lfp}(\Phi_{\mathcal{R}})$ and $\mathtt{gfp}(\Phi_{\mathcal{R}})$, obtained via the join and meet in the theorem.

They are indeed called the sets **inductively** and **coinductively defined by the rules**.

Thus indeed:

a set of rules $\Leftrightarrow$ a monotone function on a complete lattice

Next: pre-fixed points and forward closure (and dually)

What does it mean $\Phi_{\mathcal{R}}(T) \subseteq T$ (ie, set T is a pre-fixed point of $\Phi_{\mathcal{R}}$)?

As $\Phi_{\mathcal{R}}(T) = \{x \mid (S, x) \in \mathcal{R}$ for some $S \subseteq T\}$ it means:

for all rules $(S, x) \in \mathcal{R}$,
if $S \subseteq T$ (so that $x \in \Phi_{\mathcal{R}}(T)$), **then also $x \in T$.**

That is:

(i) the conclusions of each axiom is in T;

(ii) each rule whose premises are in T has also the conclusion in T.

This is precisely the 'forward' closure in previous examples.

The Fixed-point Theorem tells us that the least fixed point is the least pre-fixed point: the set inductively defined by the rules is therefore the smallest set closed forward.

For rules, the induction proof principle, in turn, says:

> for a given T,
> if for all rules $(S, x) \in \mathcal{R}$, $S \subseteq T$ implies $x \in T$
> then (the set inductively defined by the rules) $\subseteq T$.

As already seen discussing the forward closure, this is the familiar way of reasoning inductively on rules.

(the assumption "$S \subseteq T$" is the **inductive hypothesis**; the base of the induction is given by the axioms of $\mathcal{R}$)

We have recovered the **principle of rule induction**

Now the case of coinduction. Set T is a post-fixed if

$T \subseteq \Phi_{\mathcal{R}}(T)$, where $\Phi_{\mathcal{R}}(T) = \{x \mid (T', x) \in \mathcal{R} \text{ for some } T' \subseteq T\}$

This means:

for all $t \in T$ there is a rule $(S, t) \in \mathcal{R}$ with $S \subseteq T$

This is precisely the 'backward' closure

By Fixed-point Theory, the set coinductively defined by the rules is the largest set closed backward.

The coinduction proof principle reads thus (**principle of rule coinduction**):

> for a given T,
> if for all $x \in T$ there is a rule $(S, x) \in \mathcal{R}$ with $S \subseteq T$,
> then $T \subseteq$ (the set coinductive defined by the rules)

Exercise Let $\mathcal{R}$ be a set of ground rules, and suppose each rule has a non-empty premise. Show that $\mathtt{lfp}(\Phi_{\mathcal{R}}) = \emptyset$.

The examples, revisited

and continued

- the previous examples of rule induction and coinduction reduced to the fixed-point format
- example of application of bisimulation outside concurrency

Finite traces

$$\frac{P \text{ stopped}}{P \downarrow} \qquad \frac{P \xrightarrow{\mu} P' \qquad P' \downarrow}{P \downarrow}$$

As ground rules, these become:

$$\begin{aligned} \mathcal{R}_{\downarrow} \stackrel{\text{def}}{=} \ & \{(\emptyset, P) \mid P \text{ is stopped}\} \\ & \bigcup\{(\{P'\}, P) \mid P \xrightarrow{\mu} P' \text{ for some } \mu\} \end{aligned}$$

This yields the following functional:

$$\Phi_{\mathcal{R}_{\downarrow}}(T) \stackrel{\text{def}}{=} \{P \mid P \text{ is stopped, or there are } P', \mu \text{ with } P' \in T \text{ and } P \xrightarrow{\mu} P'\}$$

The sets ‘closed forward’ are the pre-fixed points of $\Phi_{\mathcal{R}_{\downarrow}}$.

Thus the smallest set closed forward and the associated proof technique become examples of inductively defined set and of induction proof principle.

ω-traces

$$\frac{P \xrightarrow{\mu} P' \qquad P' \upharpoonright}{P \upharpoonright}$$

As ground rules, this yields:

$$\mathcal{R}_{\upharpoonright} \stackrel{\text{def}}{=} \{(\{P'\}, P) \mid P \xrightarrow{\mu} P'\} \ .$$

This yields the following functional:

$$\Phi_{\mathcal{R}_{\upharpoonright}}(T) \stackrel{\text{def}}{=} \{P \mid \text{ there is } P' \in T \text{ and } P \xrightarrow{\mu} P'\}$$

Thus the sets ‘closed backward’ are the post-fixed points of $\Phi_{\mathcal{R}_{\upharpoonright}}$, and the largest set closed backward is the greatest fixed point of $\Phi_{\mathcal{R}_{\upharpoonright}}$;

Similarly, the proof technique for ω-traces is derived from the coinduction proof principle.

Finite lists ($\mathrm{finLists}$)

The rule functional (from sets to sets) is:

$$\boldsymbol{F}(\boldsymbol{T}) \stackrel{\text{def}}{=} \{\mathtt{nil}\} \cup \{\langle \boldsymbol{a} \rangle \bullet \boldsymbol{\ell} \mid \boldsymbol{a} \in \boldsymbol{A}, \boldsymbol{\ell} \in \boldsymbol{T}\}$$

$\boldsymbol{F}$ is monotone, and $\mathrm{finLists} = \mathtt{lfp}(\boldsymbol{F})$. (i.e., $\mathrm{finLists}$ is the smallest set solution to the equation $\boldsymbol{\mathcal{L}} = \mathtt{nil} + \langle \boldsymbol{A} \rangle \bullet \boldsymbol{\mathcal{L}}$).

From the induction and coinduction principles, we infer: Suppose $\boldsymbol{T} \subseteq \mathrm{finLists}$. If $\boldsymbol{F}(\boldsymbol{T}) \subseteq \boldsymbol{T}$ then $\boldsymbol{T} \subseteq \mathrm{finLists}$ (hence $\boldsymbol{T} = \mathrm{finLists}$).

Proving $\boldsymbol{F}(\boldsymbol{T}) \subseteq \boldsymbol{T}$ requires proving

- $\mathtt{nil} \in \boldsymbol{T}$;
- $\boldsymbol{\ell} \in \mathrm{finLists} \cap \boldsymbol{T}$ implies $\langle \boldsymbol{a} \rangle \bullet \boldsymbol{\ell} \in \boldsymbol{T}$, for all $\boldsymbol{a} \in \boldsymbol{A}$.

This is the same as the familiar induction technique for lists

λ-calculus

In the case of $\Downarrow$, the rules manipulate pairs of closed λ-terms, thus they act on the set $\Lambda^0 \times \Lambda^0$. The rule functional for $\Downarrow$, written $\Phi_{\Downarrow}$, is

$$\begin{aligned}\Phi_{\Downarrow}(T) \stackrel{\text{def}}{=} \; & \{(e, e') \mid e = e' = \lambda x.\, e'' \text{ , for some } e'' \} \\ & \bigcup\{(e, e') \mid e = e_1\, e_2 \text{ and} \\ & \qquad \exists\, e_0 \text{ such that } (e_1, \lambda x.\, e_0) \in T \text{ and } (e_0\{^{e_2}/_{x}\}, e') \in T\} \text{ .}\end{aligned}$$

In the case of $\Uparrow$, the rules are on Λ^0. The rule functional for $\Uparrow$ is

$$\begin{aligned}\Phi_{\Uparrow}(T) \stackrel{\text{def}}{=} \; & \{e_1\, e_2 \mid e_1 \in T, \} \\ & \bigcup\{e_1\, e_2 \mid e_1 \Downarrow \lambda x.\, e_0 \text{ and } e_0\{^{e_2}/_{x}\} \in T\}.\end{aligned}$$

Example (bisimulation outside concurrency)

Problem: reason about equality on infinite lists (streams), more generally on coinductively defined sets

Objects may be ‘infinite’, induction may not be applicable

We can prove equalities adapting the idea of bisimulation.

The coinductive definition tells us what can be observed

An LTS for lists:

$$\langle a \rangle \bullet s \xrightarrow{a} s$$

$\sim$: the resulting bisimulation

Lemma On finite/infinite lists, $s = t$ if and only if $s \sim t$.

Of course it is not necessary to define an LTS from lists.

We can directly define a kind of bisimulation on lists, as follows:

A relation $\mathcal{R}$ on lists is a **list bisimulation** if whenever $(s, t) \in \mathcal{R}$ then

1. $s = \texttt{nil}$ implies $t = \texttt{nil}$;
2. $s = \langle a \rangle \bullet s'$ implies there is t' such that $t = \langle a \rangle \bullet t'$ and $(s', t') \in \mathcal{R}$

Then **list bisimilarity** as the union of all list bisimulations.

To see how natural is the bisimulation method on lists, consider the following characterisation of equality between lists:

$$\frac{}{\mathtt{nil} = \mathtt{nil}} \qquad \frac{s_1 = s_2 \qquad a \in A}{\langle a \rangle \bullet s_1 = \langle a \rangle \bullet s_2}$$

The inductive interpretation of the rules gives us equality on finite lists, as the least fixed point of the corresponding rule functional.

The coinductive interpretation gives us equality on finite-infinite lists, and list bisimulation as associated proof technique.

To see this, it suffices to note that the post-fixed points of the rule functional are precisely the list bisimulations; hence the greatest fixed point is list bisimilarity and, by the previous Lemma, it is also the equality relation.

The coinduction/bisimulation proof method on lists

$$f : A \to A$$

$$\begin{aligned} \texttt{map}\ f\ \texttt{nil} &= \texttt{nil} \\ \texttt{map}\ f\ (\langle a \rangle \bullet s) &= \langle f(a) \rangle \bullet \texttt{map}\ f\ s \end{aligned}$$

$$\texttt{iterate}\ f\ a = \langle a \rangle \bullet \texttt{iterate}\ f\ f(a)$$

Thus $\texttt{iterate}\ f\ a$ builds the infinite list

$$\langle a \rangle \bullet \langle f(a) \rangle \bullet \langle f(f(a)) \rangle \bullet \ldots$$

For all $a \in A$: $\quad \texttt{map}\ f\ (\texttt{iterate}\ f\ a) = \texttt{iterate}\ f\ f(a)$

An LTS for lists: $\quad \langle a \rangle \bullet s \xrightarrow{a} s$

Proof

$$\mathcal{R} \stackrel{\text{def}}{=} \{(\texttt{map}\ f\ (\texttt{iterate}\ f\ a), \texttt{iterate}\ f\ f(a)) \mid a \in A\}$$

is a bisimulation. Let $(P, Q) \in \mathcal{R}$, for $P \stackrel{\text{def}}{=} \texttt{map}\ f\ (\texttt{iterate}\ f\ c)$

$Q \stackrel{\text{def}}{=} \texttt{iterate}\ f\ f(c)$

Applying the definitions of `iterate`, and of LTS

$$\begin{array}{rcl} Q & = & \langle f(c) \rangle \bullet \texttt{iterate}\ f\ f(f(c)) \\ & \xrightarrow{f(c)} & \texttt{iterate}\ f\ f(f(c)) \stackrel{\text{def}}{=} Q'. \end{array}$$

Similarly,
$$\begin{array}{rcl} P & = & \texttt{map}\ f\ \langle c \rangle \bullet (\texttt{iterate}\ f\ f(c)) \\ & = & \langle f(c) \rangle \bullet \texttt{map}\ f\ (\texttt{iterate}\ f\ f(c)) \\ & \xrightarrow{f(c)} & \texttt{map}\ f\ (\texttt{iterate}\ f\ f(c)) \\ & \stackrel{\text{def}}{=} & P' \end{array}$$

We have $P' \ \mathcal{R}\ Q'$, as $f(c) \in A$.

Done (we have showed that P and Q have a single transition, with same labels, and with derivatives in $\mathcal{R}$)

Other induction and coinduction principles

- justification from fixed-point theory
- recursion and corecursion
- enhancements of the principles

Mathematical induction

The rules (on the set $\{0, 1, \ldots\}$ of natural numbers or any set containing the natural numbers) are:

$$\frac{}{0} \qquad \frac{n}{n+1} \quad \text{(for all } n \geq 0\text{)}$$

The natural numbers: the least fixed point of a rule functional.

Principle of rule induction: if a property on the naturals holds at 0 and, whenever it holds at n, it also holds at $n+1$, then the property is true for all naturals.

This is the ordinary **mathematical induction**

A variant induction on the natural numbers: the inductive step assumes the property at all numbers less than or equal to n

$$\frac{}{\mathbf{0}} \qquad\qquad \frac{\mathbf{0}, \mathbf{1}, \ldots, n}{n+\mathbf{1}} \text{ (for all } n \geq \mathbf{0}\text{)}$$

These are the ground-rule translation of this (open) rule, where S is a property on the natural numbers:

$$\frac{i \in S\,, \;\; \forall\, i < j}{j \in S}$$

Well-founded induction

Given a well-founded relation $\mathcal{R}$ on a set X, and a property T on X, to show that $X \subseteq T$ (the property T holds at all elements of X), it suffices to prove that, for all $x \in X$: if $y \in T$ for all y with $y \mathrel{\mathcal{R}} x$, then also $x \in T$.

mathematical induction, structural induction can be seen as special cases

Well-founded induction is indeed the natural generalisation of mathematical induction to sets and, as such, it is frequent to find it in Mathematics and Computer Science.

Example: proof of a property reasoning on the lexicographical order on pairs of natural numbers

We can derive well-founded induction from fixed-point theory in the same way as we did for rule induction.

In fact, we can reduce well-founded induction to rule induction taking as rules, for each $x \in X$, the pair (S, x) where S is the set $\{y \mid y \mathcal{R} x\}$ and $\mathcal{R}$ the well-founded relation.

Note that the set inductively defined by the rules is precisely X; that is, any set equipped with a well-founded relation is an inductive set.

Transfinite induction

The extension of mathematical induction to ordinals

Transfinite induction says that to prove that a property T on the ordinals holds at all ordinals, it suffices to prove, for all ordinals α: if $\beta \in T$ for all ordinals $\beta < \alpha$ then also $\alpha \in T$.

In proofs, this is usually split into three cases:

(i) $0 \in T$;

(ii) for each ordinal α, if $\alpha \in T$ then also $\alpha + 1 \in T$;

(iii) for each limit ordinal β, if $\alpha \in \mathcal{T}$ for all $\alpha < \beta$ then also $\beta \in T$.

Transfinite induction acts on the ordinals, which form a proper class rather than a set.

As such, we cannot derive it from the fixed-point theory presented.

However, in practice, transfinite induction is used to reason on sets, in cases where mathematical induction is not sufficient because the set has ’too many’ elements.

In these cases, in the transfinite induction each ordinal is associated to an element of the set. Then the $<$ relation on the ordinals is a well-founded relation on a set, so that transfinite induction becomes a special case of well-founded induction on sets.

Another possibility: lifting the theory of induction to classes.

Other examples

Structural induction

Induction on derivation proofs

Transition induction

...

Function definitions by recursion and corecursion

One often finds functions defined by means of systems of equations. Such definitions may follow the schema of **recursion** or **corecursion**.

In a definition by *recursion* the domain of the function is an inductive set.

Examples on the well-founded set of the natural numbers: the factorial function

$$f(0) = 1 \qquad f(n+1) = n \times f(n)$$

An example of structural recursion is the function f that defines the number of λ-abstractions in a λ-term:

$$f(x) = 0 \qquad f(\lambda x.\, e) = 1 + f(e) \qquad f(e\, e') = f(e) + f(e')$$

It is possible to define patterns of equations for well-founded recursion, and prove that whenever the patterns are respected the functions specified exist and are unique. The proof makes use of well-founded induction twice, to prove that such functions exist and to prove its unicity

a function defined by **corecursion** produces an element of a coinductive set.

An equation for a corecursive function specifies the immediate observables of the element returned by the function

for instance, if the element is an infinite list, the equation should tell us specify the head of the list.

Examples are the definitions of the functions `map`, `iterate`

As in the case of recursion, so for corecursion one can produce general equation schemata, and prove that any system of equations satisfying the schemata defines a unique function (or unique functions, in case of mutually recursive equations)

Enhancements of the principles

Theorem Let F be a monotone endofunction on a complete lattice L, and y a post-fixed point of F (i.e., $y \leq F(y)$). Then

$$\mathtt{gfp}(F) = \bigcup\{x \mid x \leq F(x \cup y)\}$$

principle of coinduction up-to $\cup$:

Let F be a monotone endofunction on a complete lattice,
and suppose $y \leq F(y)$;
then $x \leq F(x \cup y)$ implies $x \leq \mathtt{gfp}(F)$.

Theorem Let F be a monotone endofunction on a complete lattice L, and $\bullet : L \times L \to L$ an associative function such that:

1. for all $x, y, x', y' \in L$, whenever both $x \leq F(x')$ and $y \leq F(y')$, then $x \bullet y \leq F(x' \bullet y')$;
2. for all x with $x \leq F(x)$ we have both $x \leq x \bullet \mathtt{gfp}(F)$ and $x \leq \mathtt{gfp}(F) \bullet x$.

Then

$$\mathtt{gfp}(F) = \bigcup \{x \mid x \leq F(\mathtt{gfp}(F) \bullet x \bullet \mathtt{gfp}(F))\}$$

principle of coinduction up-to $\mathtt{gfp}$:

Let F be a monotone endofunction on a complete lattice L, and $\bullet : L \times L \to L$ an associative function for which the assumptions (1) and (2) of Theorem above hold; then $x \leq F(\mathtt{gfp}(F) \bullet x \bullet \mathtt{gfp}(F))$ implies $x \leq \mathtt{gfp}(F)$.

Back to bisimulation

- bisimilarity as a fixed point

Bisimulation as a fixed-point

Definition Consider the following function $F_\sim : \wp(Pr \times Pr) \to \wp(Pr \times Pr)$. $F_\sim(\mathcal{R})$ is the set of all pairs (P, Q) s.t.:

1. $\forall \mu, P'$ s.t. $P \xrightarrow{\mu} P'$, then $\exists Q'$ such that $Q \xrightarrow{\mu} Q'$ and $P' \mathcal{R} Q'$;
2. $\forall \mu, Q'$ s.t. $Q \xrightarrow{\mu} Q'$, then $\exists P'$ such that $P \xrightarrow{\mu} P'$ and $P' \mathcal{R} Q'$.

Proposition We have:

- $F_\sim$ is monotone;
- $\mathcal{R}$ is a bisimulation iff $\mathcal{R} \subseteq F_\sim(\mathcal{R})$;
- $\sim = \mathtt{gfp}(F_\sim)$.

Least and greatest fixed points:

approximations and constructive proofs

– objective: derive the reading (3) of inductive/coinductive sets
(via iteration schema)

Continuity and cocontinuity

The proof of the Fixed-point Theorem is not constructive

We show now constructive proofs, by means of iterative schemata.

Pros:

- approximating/computing least fixed points and greatest fixed points. (at the heart of the algorithms used in tools)
- an alternative way for reasoning

Cons:

- requires properties on functions (continuity/cocontinuity) stronger than monotonicity.

Abbreviations: $\bigcup_i \alpha_i$ for $\bigcup_i \{\alpha_i\}$, and $\bigcup_i F(\alpha_i)$ for $\bigcup_i \{F(\alpha_i)\}$; similarly for $\bigcap_i \alpha_i$ and $\bigcap_i F(\alpha_i)$.

Definition An endofunction on a complete lattice is:

- **continuous** if for all sequences $T_0, T_1 \ldots$ of increasing points in the lattice (i.e., $T_i \subseteq T_{i+1}$, for $i \geq 0$) we have $F(\bigcup_i T_i) = \bigcup_i F(T_i)$.
- **cocontinuous** if for all sequences $T_0, T_1 \ldots$ of decreasing points in the lattice (i.e., $T_{i+1} \subseteq T_i$, for $i \geq 0$) we have $F(\bigcap_i T_i) = \bigcap_i F(T_i)$.

Example: the complete lattice made of the integers plus the points ω and $-\omega$, with the ordering $-\omega \leq n \leq \omega$ for all n.

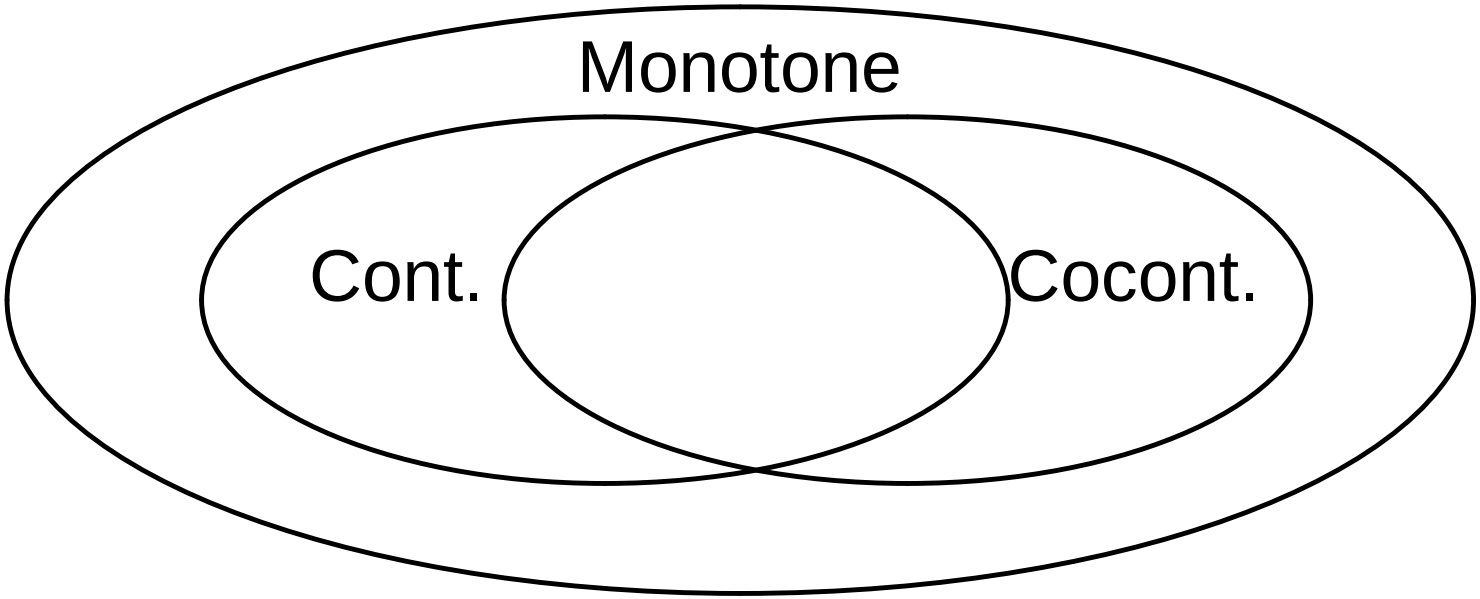
Take F with: $F(n) = n + 1$, $F(\omega) = \omega$, $F(-\omega) = -\omega$

Then $F(\cup\{3, 4, 6\}) = F(6) = 7 = \cup\{4, 5, 7\} = \cup\{F(3), F(4), F(6)\}$

For the increasing sequence of the positive integers, we have
$F(\cup_i n_i) = F(\omega) = \omega = \cup_i n_{i+1} = \cup_i F(n_i)$.

Dually, for the decreasing sequence of the negative integers, we have
$F(\cap_i - n_i) = -\omega = \cap_i F(-n_i)$.

Exercise If F is cocontinuous (or continuous), then it is also monotone.
(Hint: Take $x \geq y$, and the sequence $x, y, y, y, \ldots$.) $\square$



$F^n(x)$ indicates the n-th iteration of F starting from the point x:

$$\begin{aligned} F^0(x) &\stackrel{\text{def}}{=} x \\ F^{n+1}(x) &\stackrel{\text{def}}{=} F(F^n(x)) \end{aligned}$$

Then we set:

$$\begin{aligned} F^{\cap\omega}(x) &\stackrel{\text{def}}{=} \bigcap_{n\geq 0} F^n(x) \\ F^{\cup\omega}(x) &\stackrel{\text{def}}{=} \bigcup_{n\geq 0} F^n(x) \end{aligned}$$

Theorem [Continuity/Cocontinuity] Let F be an endofunction on a complete lattice, in which $\perp$ and $\top$ are the bottom and top elements.

- If F is continuous, then $\mathtt{lfp}(F) = F^{\cup\omega}(\perp)$;
- if F is cocontinuous, then $\mathtt{gfp}(F) = F^{\cap\omega}(\top)$.

$F^0(\perp), F^1(\perp), \ldots$ is increasing, $F^0(\top), F^1(\top), \ldots$ is decreasing.

lfp and gfp of F are the join and meet of the two sequences.

Exercise Prove the Continuity/Cocontinuity Theorem. (Hint: Referring to the second part, first show that $F^{\cap\omega}$ is a fixed point, exploiting the definition of cocontinuity; then show that it is the greatest fixed point, exploiting the definition of meet.)

If F is not cocontinuous, and only monotone, we only have $\mathtt{gfp}(F) \leq F^{\cap\omega}(\top)$. The converse need not hold

However, when $F^{\cap\omega}(\top)$ is a fixed point, then surely it is the greatest fixed point.

Example Let L be the set of negative integers plus the elements $-\omega$ and $-(\omega+1)$, with the expected ordering $-n \geq -\omega \geq -(\omega+1)$, for all n. Let now F be the following function on L:

$$\begin{array}{rcl} F(-n) & = & -(n+1) \\ F(-\omega) & = & -(\omega+1) \\ F(-(\omega+1)) & = & -(\omega+1) \end{array}$$

The top and bottom elements are -1 and $-(\omega+1)$. Function F is monotone but not cocontinuous, and we have $F^{\cap\omega}(-1) = -\omega$ and $\mathtt{gfp}(F) = -(\omega+1)$.

Having only monotonicity, to reach the greatest fixed point using induction, we need to iterate over the transfinite ordinals.

(The dual statement, for continuity and least fixed points, also holds.)

Theorem Let F be a monotone endofunction on a complete lattice L, and define $F^{\lambda}(\top)$, where λ is an ordinal, as follows:

$$\begin{array}{rcll} F^{0}(\top) & \stackrel{\text{def}}{=} & \top & \\ F^{\lambda+1}(\top) & \stackrel{\text{def}}{=} & F(F^{\lambda}(\top)) & \text{for successor ordinals} \\ F^{\lambda}(\top) & \stackrel{\text{def}}{=} & F(\bigcap_{\beta<\lambda} F^{\beta}(\top)) & \text{for limit ordinals} \end{array}$$

and then $F^{\infty}(\top) \stackrel{\text{def}}{=} \bigcap_{\lambda} F^{\lambda}(\top)$. We have:

$$F^{\infty}(\top) = \mathtt{gfp}(F).$$

Continuity and cocontinuity, for rules

The functional given by a set of rules need not be continuous or cocontinuous.

Example: Take the rule

$$\frac{a_1 \quad \dots \quad a_n \quad \dots}{a}$$

and call ϕ its functional, $T_n = \{a_1, \dots, a_n\}$.
Then $a \in \phi(\bigcup_n T_n)$, but $a \notin \bigcup_n \phi(T_n)$ (hence ϕ not continuous)

We can recover continuity and cocontinuity for rule functionals adding some conditions.

Definition A set $\mathcal{R}$ of rules is **finite in the premises**, briefly **FP**, if for each rule $(S, x) \in \mathcal{R}$ the premise set S is finite.

Exercise Show that if the set of rules $\mathcal{R}$ is FP, then $\Phi_{\mathcal{R}}$ is continuous; conclude that $\mathtt{lfp}(\Phi_{\mathcal{R}}) = \Phi_{\mathcal{R}}^{\bigcup\omega}(\emptyset)$.

FP does not work for cocontinuity

Example. $X = \{b\} \cup \{a_1, \ldots, a_n, \ldots\}$, and rules $\dfrac{a_i}{b} \quad \forall\, i$

call Φ the corresponding rule functional.

Thus $\Phi(T) = \{b\}$ if there is i with $a_i \in T$, otherwise $\Phi(T) = \emptyset$.

Take the sequence of decreasing sets $T_0, \ldots, T_n, \ldots$, where

$$T_i \stackrel{\text{def}}{=} \{a_j \mid j \geq i\}$$

We have $\Phi(\bigcap_n T_n) = \emptyset$, but $\bigcap_n \Phi(T_n) = \{b\}$.

To obtain cocontinuity we need some finiteness conditions on the conclusions of the rules (rather than on the premises as for continuity).

Definition A set of rules $\mathcal{R}$ is **finite in the conclusions**, briefly FC, if for each x, the set $\{S \mid (S, x) \in \mathcal{R}\}$ is finite (i.e., there is only a finite number of rules whose conclusion is x)

Each premise set S may itself be infinite

Theorem If a set of rules $\mathcal{R}$ is FC, then $\Phi_{\mathcal{R}}$ is cocontinuous.

Exercise Prove the theorem above.

Corollary If a set of rules $\mathcal{R}$ on X is FC, then $\mathtt{gfp}(\Phi_{\mathcal{R}}) = \Phi_{\mathcal{R}}^{\cap\omega}(X)$.

Without FC, and therefore without cocontinuity, we have nevertheless $\mathtt{gfp}(\Phi_{\mathcal{R}}) \subseteq \Phi_{\mathcal{R}}^{\cap\omega}(X)$.

With the FP or FC hypothesis we are thus able of applying the Continuity/Cocontinuity Theorem.

For FP and continuity, the theorem tells us that given some rules $\mathcal{R}$, the set inductively defined by $\mathcal{R}$ can be obtained as the limit of the increasing sequence of sets

$$\emptyset, \Phi_{\mathcal{R}}(\emptyset), \Phi_{\mathcal{R}}(\Phi_{\mathcal{R}}(\emptyset)), \Phi_{\mathcal{R}}(\Phi_{\mathcal{R}}(\Phi_{\mathcal{R}}(\emptyset))), \ldots.$$

That is, we construct the inductive set thus:

- start with the empty set
- add the conclusions of the axioms in $\mathcal{R}$ ($\Phi_{\mathcal{R}}(\emptyset)$),
- repeatedly add elements following the inference rules in $\mathcal{R}$ in a 'forward' manner

This corresponds to the usual constructive way of interpreting inductively a bunch of rules

As usual, the case for coinductively defined sets is dual.

The iterative reading, for the finite-trace example

Exercise The rules for finite traces:

$$\begin{aligned}\mathcal{R}_{\lfloor} \stackrel{\text{def}}{=} & \{(\emptyset, P) \mid P \text{ is stopped}\} \\ & \bigcup\{(\{P'\}, P) \mid P \xrightarrow{\mu} P' \text{ for some } \mu\}\end{aligned}$$

Show that:

- $P \in \Phi^n_{\mathcal{R}_{\lfloor}}(\emptyset)$, for $0 \leq n$, if and only if there are $0 \leq m \leq n$, processes $P_0, \ldots, P_m$, and actions $\mu_1, \ldots, \mu_m$ with $P = P_0$ and such that $P_0 \xrightarrow{\mu_1} P_1 \ldots \xrightarrow{\mu_m} P_m$ and P_m is stopped.

At step 0 we have the empty set; then at step 1 we add the stopped processes; at step 2 we add the processes that have a stopped derivative; and so on.

In applications in which the set of all processes is finite, the sequence $\{\Phi^n_{\mathcal{R}_{\lfloor}}(\emptyset)\}_n$ will not increase forever

The iterative reading, for the ω-trace example

Recall that the ground rules are:

$$\mathcal{R}_{\restriction} \stackrel{\text{def}}{=} \{(\{P'\}, P) \mid P \xrightarrow{\mu} P'\} .$$

Exercise For *Pr* = all processes, show that:

- $P \in \Phi^n_{\mathcal{R}_{\restriction}}(Pr)$, for $0 \leq n$, if and only if there are processes $P_0, \ldots, P_n$ with $P = P_0$ and such that $P_0 \xrightarrow{\mu} P_1 \ldots \xrightarrow{\mu} P_n$.

In the sequence

$$\Phi^0_{\mathcal{R}_{\restriction}}(Pr), \Phi^1_{\mathcal{R}_{\restriction}}(Pr), \Phi^2_{\mathcal{R}_{\restriction}}(Pr), \cdots$$

at step 0 we have the set *Pr* of all processes; at step 1 we remove the processes that do not have a μ-derivative; at step 2 the processes that cannot perform 2 consecutive μ-transitions; and so on.

If the set of processes is finite, the sequence will not decrease forever

Approximants of bisimilarity

Here is a natural definition of approximants of bisimilarity, where *Pr* = the states of an LTS

- $\sim_0 \stackrel{\text{def}}{=} Pr \times Pr$;
- $P \sim_{n+1} Q$, for $n \geq 0$, if for all μ:
  1. for all P' with $P \xrightarrow{\mu} P'$, there is Q' such that $Q \xrightarrow{\mu} Q'$ and $P' \sim_n Q'$;
  2. the converse, i.e., for all Q' with $Q \xrightarrow{\mu} Q'$, there is P' such that $P \xrightarrow{\mu} P'$ and $P' \sim_n Q'$;
- $\sim_\omega \stackrel{\text{def}}{=} \bigcap_{n \geq 0} \sim_n$.

At stage n, we check transitions up to depth n.

There is an exact correspondence with the the sequence

$$F_\sim^0(Pr \times Pr), F_\sim^1(Pr \times Pr), F_\sim^2(Pr \times Pr), \cdots$$

where $F_\sim$ is the functional of bisimilarity

Recall that $F_\sim(\mathcal{R})$ is the set of all pairs (P, Q) s.t.:

1. $\forall \mu, P'$ s.t. $P \xrightarrow{\mu} P'$, then $\exists Q'$ such that $Q \xrightarrow{\mu} Q'$ and $P' \mathcal{R} Q'$;
2. $\forall \mu, Q'$ s.t. $Q \xrightarrow{\mu} Q'$, then $\exists P'$ such that $P \xrightarrow{\mu} P'$ and $P' \mathcal{R} Q'$.

Exercise

1. $\sim_0, \ldots, \sim_n, \ldots$ is a decreasing sequence of relations.
2. For all $0 \leq n < \omega$, we have $\sim_n = F_\sim^n(Pr \times Pr)$, and $\sim_\omega = F_\sim^{\bigcap\omega}(Pr \times Pr)$, where $F_\sim^n$ and $F_\sim^{\bigcap\omega}$ are the iterations of $F_\sim$ following the definitions used in the Cocontinuity Theorem

NB: Approximants can also be usefully employed to prove non-bisimilarity results

Bisimulation and cocontinuity

A counterexample to $\sim\ =\ \sim_\omega$. Take the states and transitions:

$$\begin{array}{lcl} a^0 & \stackrel{\text{def}}{=} & 0 \\ a^\omega & \xrightarrow{a} & a^\omega \\ a^n & \xrightarrow{a} & a^{n-1} \quad \text{for } n \geq 1 \end{array}$$

Now let P, Q be states with transitions

$$P \xrightarrow{a} a^n \quad \text{for all } n \geq 0$$

and

$$\begin{array}{ll} Q \xrightarrow{a} a^n & \text{for all } n \geq 0 \\ Q \xrightarrow{a} a^\omega & \end{array}$$

$P \sim_n Q$ for all n (simple induction), hence also $P \sim_\omega Q$.

$P \not\sim Q$, as $Q \xrightarrow{a} a^\omega$ can only be matched by $P \xrightarrow{a} a^n$, for some n

Exercise Show formally that the functional $F_\sim$ of bisimilarity is not cocontinuous.

A sufficient condition: finite branching

We can obtain $\sim$ by iteration over the natural numbers if we add some finiteness hypothesis on the branching structure of the LTS.

An LTS is **finitely-branching** if for each process the set of its (immediate) derivatives is finite.

Theorem On finitely-branching LTSs, $\sim \; = \; \sim_\omega$.

The thereom follows from the following exercise and the Cocontibuity Theorem.

Exercise Check that under the finitely-branching hypothesis the functional $F_\sim$ is cocontinuous.

A direct proof is useful to understand the hypothesis

Proof The inclusion $\sim \subseteq \sim_\omega$ is easy: one proves that $\sim \subseteq \sim_n$ for all n, using the fact that $\sim$ is a bisimulation

(or, using the fact that $\sim$ is a fixed point of $F_\sim$, monotonicity of $F_\sim$, and $\sim_{n+1} = F_\sim(\sim_n)$; we can also directly derive it from fixed-point theory).

Now the converse. We show that the set

$$\mathcal{R} \stackrel{\text{def}}{=} \{(P, Q) \mid P \sim_\omega Q\}$$

is a bisimulation. Take $(P, Q) \in \mathcal{R}$ and suppose $P \xrightarrow{\mu} P'$.

$\forall\, n$, as $P \sim_{n+1} Q$, $\exists\, Q_n$ s.t. $Q \xrightarrow{\mu} Q_n$ and $P \sim_n Q_n$.

As the LTS is finitely-branching, the set $\{Q_i \mid Q \xrightarrow{\mu} Q_i\}$ is finite.

Hence $\exists\, Q_i$ s.t. $P' \sim_n Q_i$ for infinitely many n.

As $\{\sim_n\}_n$ is decreasing with n, we have $P' \sim_n Q_i \;\; \forall\, n$.

Hence $P' \sim_\omega Q_i$ and $(P', Q_i) \in \mathcal{R}$. $\square$

Algorithms for bisimilarity

The stratification of bisimilarity given by continuity is also the basis for **algorithms** for mechanically checking bisimilarity and for minimisation of the state-space of a process

These algorithms work on processes that are **finite-state** (ie, each process has only a finite number of possible derivaties)

They proceed by progressively refining a partition of all processes

In the initial partition, all processes are in the same set

Bisimulation: P-complete **[Alvarez, Balcazar, Gabarro, Santha, '91]**

With m transitions, n states:

O(m log n) time and O(m + n) space **[Paige, Tarjan, '87]**

Trace equivalence, testing: PSPACE-complete

[Kannelakis, Smolka, '90; Huynh, Tian, 95]

Other views on the meaning of induction and coinduction

- derivation proofs
  (cf: the informal reading (1) of inductive and coinductive sets)
- games

Proof tree interpretations

A set of ground rules is used to derive elements, via proof trees

The example of lists. The rules are $\dfrac{}{\texttt{nil}}$ $\quad \dfrac{s \qquad a}{\langle a \rangle \bullet s}$ $\quad (\forall s \in X, a \in A)$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\texttt{nil}}}{\langle b \rangle \bullet \texttt{nil}}}{\langle a \rangle \bullet \langle b \rangle \bullet \texttt{nil}}}{\langle a \rangle \bullet \langle a \rangle \bullet \langle b \rangle \bullet \texttt{nil}} \qquad \dfrac{\dfrac{\dfrac{\dots}{\langle a \rangle \bullet \langle b \rangle \bullet \langle a \rangle \bullet \langle b \rangle \bullet \dots}}{\langle b \rangle \bullet \langle a \rangle \bullet \langle b \rangle \bullet \langle a \rangle \bullet \langle b \rangle \bullet \dots}}{\langle a \rangle \bullet \langle b \rangle \bullet \langle a \rangle \bullet \langle b \rangle \bullet \langle a \rangle \bullet \langle b \rangle \bullet \dots}$$

(in general — but not with lists — a node of a tree may have several children)

$\langle a \rangle \bullet \langle a \rangle \bullet \langle b \rangle \bullet \texttt{nil}$ is both in the inductive and in the coinductive set

$\langle a \rangle \bullet \langle b \rangle \bullet \langle a \rangle \bullet \langle b \rangle \bullet \dots$ in only in the coinductive set

What is the difference between induction and coinduction on the meaning of ‘correct’ proof tree?

Trees (informally)

The set of **trees over X** is the set of all trees, possibly infinite both in depth and in breadth, in which each node is labelled with an element from the set X and, moreover, the labels of the children of a node are pairwise distinct.

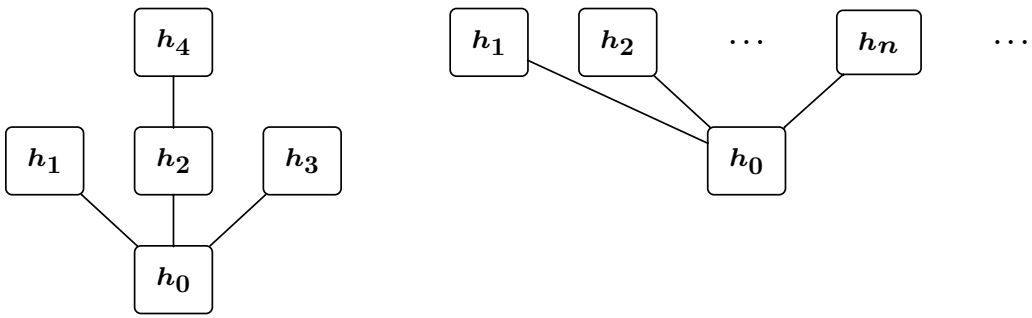
If $\mathcal{T}$ is such a tree, then the **root** of $\mathcal{T}$ is the only node without a parent.

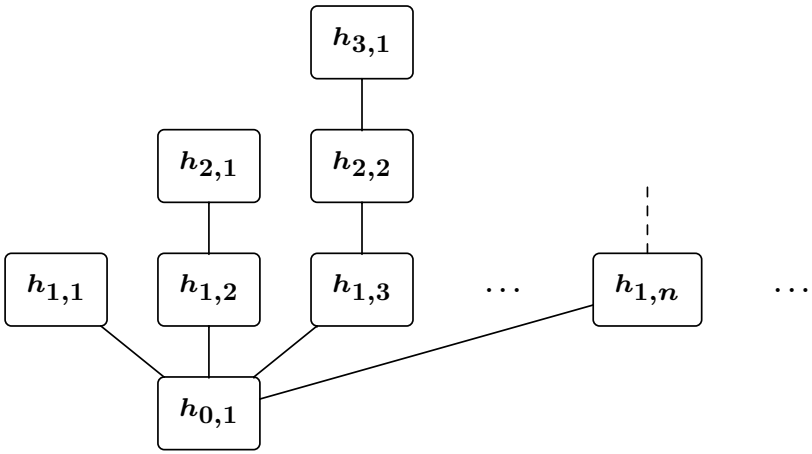
Let $\mathcal{R}$ be a set of ground rules.

A tree $\mathcal{T}$ is **a proof tree for $x \in X$ under $\mathcal{R}$** if x is the label of the root of $\mathcal{T}$ and, for each node h with label y, if S is the set of the labels of all children of h, then (S, y) is a rule in $\mathcal{R}$.

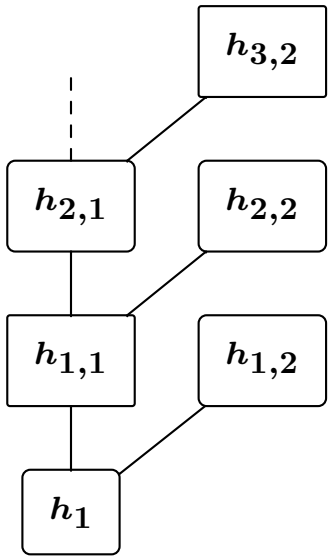
A tree is *non-well-founded* if it has paths of infinite length. It is *well-founded* otherwise.

Some well-founded trees:





A non-well-founded tree:



Theorem $x \in \mathtt{lfp}(\Phi_{\mathcal{R}})$ iff there is a well-founded proof tree for x under $\mathcal{R}$.

Proof: reason on the approximants

With FP hypothesis, each node only has finitely many children, and therefore a well-founded proof tree has a finite height (hence it is finite)

In the examples of traces, λ-calculus, lists, the rules are FP, hence the inductive objects are those with a ‘finite derivation proof’.

Without FP, a well-founded proof tree need not have a finite height.

Theorem $x \in \mathtt{gfp}(\Phi_{\mathcal{R}})$ iff there is a proof tree for x under $\mathcal{R}$.

Proof First, the direction from left to right.

If $x \in \mathtt{gfp}(\Phi_{\mathcal{R}})$, then x is in some post-fixed point of $\Phi_{\mathcal{R}}$; that is, there is T with $x \in T$ and $T \subseteq \Phi_{\mathcal{R}}(T)$.

Now, as $T \subseteq \Phi_{\mathcal{R}}(T)$, by definition of $\Phi_{\mathcal{R}}$, for each $y \in T$ there is at least one rule (S, y) in $\mathcal{R}$ with $S \subseteq T$; we pick one of these rules and call it R_y.

The proof tree for x is defined as follows. The root is x. The children of a node y in the tree are the nodes $y_1, \ldots, y_n$ that form the premise of the rule R_y chosen for y.

Theorem $x \in \mathtt{gfp}(\Phi_{\mathcal{R}})$ iff there is a proof tree for x under $\mathcal{R}$.

Proof Now the converse direction (right to left).

Suppose there is a proof tree for x.

Let T be the set of all the (labels of) nodes in the tree. We show that T is a post-fixed point of $\Phi_{\mathcal{R}}$.

We have to show that any $y \in T$ is in $\Phi_{\mathcal{R}}(T)$.

If $y \in T$ then there is a node in the tree that is labelled y.

Let $\{y_1, \ldots, y_n\}$ be the set of the labels of the children of such node.

By definition of proof tree, $(\{y_1, \ldots, y_n\}, y)$ is a rule in $\mathcal{R}$ and, by definition of T, we have $(\{y_1, \ldots, y_n\} \subseteq T$.

Hence $y \in \Phi_{\mathcal{R}}(T)$.

Game interpretations

A game-theoretic characterisation of induction and coinduction, exploiting some of the ideas in the ‘proof-tree’ presentation

Consider a set $\mathcal{R}$ of ground rules (on X).

A **game in $\mathcal{R}$** has:

- two players (the verifier V and the refuter R)
- an element $x_0 \in X$

V tries to show that a proof tree for x_0 exists; R tries to dispute that

A play

Thus **a play for $\mathcal{R}$ and x_0** is a sequence

$$x_0, S_0, \ldots, x_n, S_n, \ldots$$

which can be finite or infinite.

It goes thus:

- V chooses a set S_0 s.t. $(S_0, x_0) \in \mathcal{R}$ (i.e., x_0 can be derived from S_0)
- R chooses an element $x_1 \in S_0$ (thus challenging V to continue)
- V has to find a set S_1 with $(S_1, x_1) \in \mathcal{R}$
- R picks $x_2 \in S_1$
- and so on.

Example from the λ-calculus

The ground rules $\mathcal{R}$ for the divergence predicate $\Uparrow$ of the λ-calculus are

$$\frac{e}{e\ e'} \qquad \frac{e}{e_1\ e_2} \text{ with } e_1 \Downarrow \lambda x.\, e_0 \text{ for some } e_0 \text{ with } e_0\{e_2/x\} = e$$

For $e_1 = \lambda x.\, xx$, a play is

$$e_1\ e_1,\ \{e_1\ e_1\},\ e_1\ e_1,\ \ldots$$

For $e_2 = \lambda x.\, xxx$, a play is

$$e_2\ e_2,\ \{(e_2\ e_2)\ e_2\},\ (e_2\ e_2)\ e_2,\ \{e_2\ e_2\},\ \ldots$$

Both plays, in the coinductive game, represent win for V. A finite play for $e_2\ e_2$ is

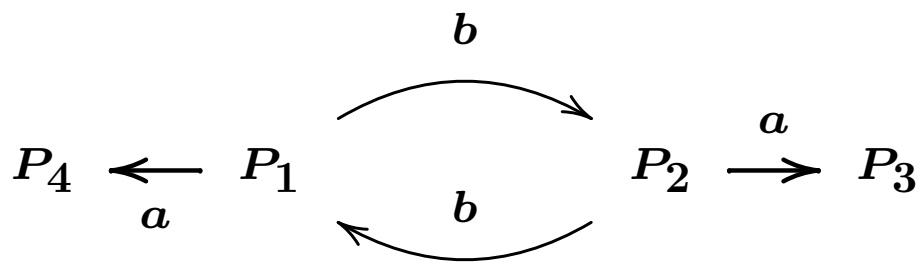
$$e_2\ e_2,\ \{e_2\},\ e_2$$

and it is a win for R.

Example with finite traces

The ground rules $\mathcal{R}_{\downharpoonleft}$ for the finite-trace predicate $\downharpoonleft$ are

$$\frac{}{P} \quad \text{with } P \text{ stopped} \qquad\qquad \frac{P'}{P} \quad \text{with } P \xrightarrow{\mu} P'$$



A play for $\mathcal{R}_{\downharpoonleft}$ and P_1 is

$$P_1,\ \{P_2\},\ P_2,\ \{P_1\},\ P_1, \ldots$$

where V follows the b-transitions; another play is

$$P_1,\ \{P_2\},\ P_2,\ \emptyset$$

where V follows an a-transition. The latter play is a win for V. In the inductive game, the first play is a win for R.

Inductive vs coinductive games

The game is **finite** if at some point one of the players is unable to make the move; then the other player wins.

(e.g., V's last move was the empty set $\emptyset$; R's last move was an element x that does not appear in conclusions of the rules $\mathcal{R}$)

An **infinite** game:

- in the inductive world it is a win for R
  (with induction, proofs should must be well-founded)
- in the coinductive world it is a win for V
  (with coinduction, non-well-founded paths in proof trees are allowed)

$\mathcal{G}^{\mathtt{ind}}(\mathcal{R}, x_0)$: the inductive game.

$\mathcal{G}^{\mathtt{coind}}(\mathcal{R}, x_0)$: the coinductive game.

Strategies

A **winning strategy**: a systematic way of playing that always produces a win

Definition In a game $\mathcal{G}^{\mathtt{ind}}(\mathcal{R}, x_0)$ or $\mathcal{G}^{\mathtt{coind}}(\mathcal{R}, x_0)$:
a **strategy for** V is a partial function that associates to each play

$$x_0, S_0, \ldots, x_n, S_n, x_{n+1}$$

a set S_{n+1}, with $(S_{n+1}, x_{n+1}) \in \mathcal{R}$, to be used for the next move for V; similarly, a **strategy for** R in $\mathcal{G}^{\mathtt{ind}}(\mathcal{R}, x_0)$ or $\mathcal{G}^{\mathtt{coind}}(\mathcal{R}, x_0)$ is a partial function that associates to each play

$$x_0, S_0, \ldots, x_n, S_n$$

an element $x_{n+1} \in S_n$. The strategy of a player is **winning** if that player wins every play in which he/she has followed the strategy.

The strategies for induction and coinduction can actually be history-free

Exercise Analise the winning stragegies for the previous examples

Theorem

1. $x_0 \in \mathtt{lfp}(\Phi_{\mathcal{R}})$ iff player V has a winning strategy in the game $\mathcal{G}^{\mathtt{ind}}(\mathcal{R}, x_0)$;
2. $x_0 \in \mathtt{gfp}(\Phi_{\mathcal{R}})$ iff player V has a winning strategy in the game $\mathcal{G}^{\mathtt{coind}}(\mathcal{R}, x_0)$.

The bisimulation game

As we have seen, there is a standard construction to turn any monotone operator on a complete lattice $\wp(X)$ into a set of rules (and vice versa)

This construction gives us these rules for bisimulation:

$$\frac{\mathtt{Der}(P, Q, f, g)}{(P, Q)}$$

where

- function f maps a pair (μ, P') such that $P \xrightarrow{\mu} P'$ into a process Q' such that $Q \xrightarrow{\mu} Q'$
  Conversely, function g maps a pair (μ, Q') such that $Q \xrightarrow{\mu} Q'$ into a process P' such that $P \xrightarrow{\mu} P'$
- $\mathtt{Der}(P, Q, f, g)$ is the set of process pairs

$$\{(P', f(\mu, P')) \mid P \xrightarrow{\mu} P'\} \cup \{(g(\mu, Q'), Q') \mid Q \xrightarrow{\mu} Q'\} .$$

(With non-determinism, there may be several rules with the same conclusion)

In the resulting game interpretation, given a pair (P, Q), the verifier V chooses the functions f and g that determine the pairs $\mathtt{Der}(P, Q, f, g)$ needed in the premise.

The refuter R then picks up one of the pairs in $\mathtt{Der}(P, Q, f, g)$ to continue the game.

If functions f and g for V do not exist, then V cannot continue and R wins.

When $\mathtt{Der}(P, Q, f, g)$ is empty (which happens if both P and Q are stopped), R cannot continue and V wins.

As the game is coinductive, an infinite play represents a win for V.

Example

$$\begin{tikzcd}[row sep=small] & P_2 \\ P_1 \arrow[ur, "a"] \arrow[dr, "a"'] & \\ & P_3 \arrow[r, "b"'] & P_4 \end{tikzcd}$$

$$Q_1 \xrightarrow{a} Q_2 \xrightarrow{b} Q_3$$

A play in which V wins:

$$(P_1, Q_1),\ \{(P_2, Q_2), (P_3, Q_2)\},\ (P_3, Q_2),\ \{(P_4, Q_3)\},\ (P_4, Q_3),\ \emptyset$$

A play with a win for R:

$$(P_1, Q_1),\ \{(P_2, Q_2), (P_3, Q_2)\},\ (P_2, Q_2)$$

R has a winning strategy, which consists in following the latter play, thereby always selecting, in the first move, the pair (P_2, Q_2).

A simpler bisimulation game

We formulate the bisimulation game a bit differently, letting R move first

R first chooses a transition, say $P \xrightarrow{\mu} P'$ or $Q \xrightarrow{\mu} Q'$

then V has to find a matching derivative from Q or P

A **play for (P_0, Q_0)** in the new game is a finite or infinite sequence of pairs

$$(P_0, Q_0),\ (P_1, Q_1),\ \cdots,\ (P_i, Q_i),\ \cdots$$

Given (P_i, Q_i), the following pair (P_{i+1}, Q_{i+1}) is determined thus:

- R makes the challenge by choosing either a transition $P_i \xrightarrow{\mu} P'$ or a transition $Q_i \xrightarrow{\mu} Q'$;
- V has to answer, in the former case with a transition $Q_i \xrightarrow{\mu} Q'$, in the latter case with a transition $P_i \xrightarrow{\mu} P'$;
- the pair (P', Q') is $(i+1)$th one of the play.

If V is unable to answer, then R wins.

If this situation never occurs (at some point R cannot formulate a challenge, or the play is infinite) then V is the winner.

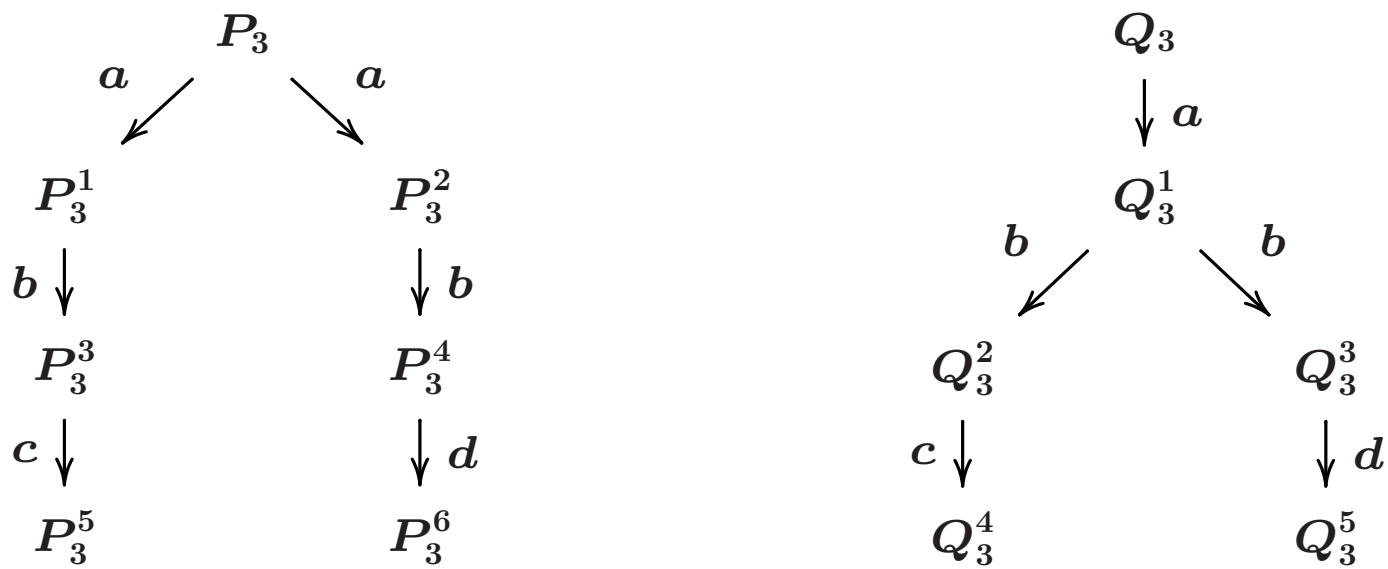
Again, we can define the notion of strategy

Theorem $P \sim Q$ if and only if V has a winning strategy for (P, Q).

Theorem $P \not\sim Q$ if and only if R has a winning strategy for (P, Q).

The game interpretation is also useful to reason about bisimulation, especially to prove non-bisimilarity results.

Example: a winning strategy for R



- The initial transition chosen by R is $P_3 \xrightarrow{a} P_3^1$.
- The only answer for V can be via the transition $Q_3 \xrightarrow{a} Q_3^1$, and the resulting pair is (P_3^1, Q_3^1).
- Now R chooses the transition $Q_3^1 \xrightarrow{b} Q_3^3$, and V has only the transition $P_3^1 \xrightarrow{b} P_3^3$, resulting in the new pair (P_3^3, Q_3^3).
- Finally, R makes the challenge on the transition $Q_3^3 \xrightarrow{d} Q_3^5$, and V cannot answer.