Viewing λ-terms through Maps

Masahiko Sato Kyoto University

Randy Pollack Harvard University

Helmut Schwichtenberg University of Munich

Takafumi Sakurai Chiba University

Version of March 30, 2014

Outline

Reminder: eigenvariable problems

- Arbitrary choice of p in INTRO: judgements may have infinitely many derivations:

$$\frac{\Gamma \text{ valid} \qquad p{:}A \in \Gamma}{\Gamma \vdash p : A} \qquad (\text{ELIM})\ \frac{\Gamma \vdash b : A {\to} B \qquad \Gamma \vdash a : A}{\Gamma \vdash b\,a : B}$$

$$(\text{INTRO})\ \frac{\Gamma, p{:}A \vdash b\{y \backslash p\} : B \qquad p \sharp b}{\Gamma \vdash \lambda y.b : A {\to} B}$$

- Cannot prove weakening of this system by structural induction.
- A solution: It is equivalent to replace INTRO with:

$$(\text{INTRO})\ \frac{\forall p\,.\, p \sharp \Gamma \;\Rightarrow\; \Gamma, p{:}A \vdash b\{y \backslash p\} : B}{\Gamma \vdash \lambda y.b : A {\to} B}$$

No arbitrary choices: judgements have at most one derivation:

So what's the problem?

- Many species of names:
  - local and global term variables, type variables, module variables, ...
  - Different kinds of substitution for different variable species.
- Proof of equivalence of the 2 relations requires equivariance and name swapping.
- The universally quantified INTRO rule isn't really syntax:
  - not formalizable in PRA.
- Not easy to mechanically derive the universally quantified rules:
  - Nominal Isabelle, which attempts to construct similar strengthened rules, sometimes fails on naturally occurring relations.
- This equivalence must be derived for every relation of interest:
  - simple reduction, parallel reduction, complete development, typing, ...

Desiderata for a formal representation of binding

- Concrete: inductively definable in (say) Coq and HOL.
  - nominal Isabelle fails: needs extensionality and quotients.
- Canonical: α -equivalence is identity.
  - McKinna and Pollack representation (from 1993) fails.
- Reasoning: structural.
  - Pure de Bruijn fails.
- Reasoning without equivariance, name swapping, special derived induction principles, etc.
  - Locally nameless and Sato canonical representations fail.

HOAS; 2-level representations ???

???

The representation of this talk makes some progress.

Intuition: *Maps* for binding

- *Map*s generalize the notion of *occurrance*.
- Maps are binary trees over 0 and 1.
- Example:
  - Occurrences of x in $(xz)(yz)$ represented by map $(10)(00)$.
  - Occurrences of z in $(xz)(yz)$ represented by map $(01)(01)$.
- λ-term $\mathsf{S} = \lambda xyz.(xz)(yz)$ is represented

$$(10\ 00)\backslash(00\ 10)\backslash(01\ 01)\backslash(\square\square\ \square\square)$$

(We drop some parentheses for readability.)

- Bound positions represented only by constant $\square$ (called *box*).

Open terms

- $\Box$ may occur unbound.
  - $\Box$ is a distinguished constant.
  - We accept $\Box$ as a term.
  - $1\backslash\Box$ represents $\lambda z.z$.
  - $0\backslash\Box$ represents $\lambda x.z$.
    - Unbound box is available for binding or substitution,
    - $1\backslash 0\backslash\Box$ represents $\lambda z.\lambda x.z$.
- Free variables may occur in terms,
  - the informal term $\lambda z.(xz)$ is written as $(0\ 1)\backslash(x\ \Box)$.
- There are no bound names or de Bruijn indices.

Well-formedness conditions needed

- Free variables cannot be bound:
  - maps can only bind $\square$,
  - $0\backslash x$ is a term, $1\backslash x$ is not a term.
  - We will show how to bind names.
- Want canonical representation: one representative per λ -term.
  - $0\backslash 1\backslash \square$ is our notation for $\lambda x.\lambda x.x$ (which equals $\lambda y.\lambda x.x$)
  - $1\backslash 1\backslash \square$ is not a term.
- Substitution: Consider the term $(0\ 1)\backslash(\square\ \square)$;
  - position $(1\ 0)$ (the red $\square$) is free,
  - substitute $(\square\ \square)$ in that position,
  - get $(0\ 1)\backslash((\square\ \square)\ \square)$ which is not a term because 0 is not a position in $(\square\ \square)$.
  - The solution: identify maps 0 and $(0\ 0)$.

Compare with other notations

- Abstraction by names (raw terms or nominal terms):
  - Binding information shared between binding occurrences and bound occurrences (shared names).
  - Substitution may require α -conversion of the base term.
- Abstraction by indexes (de Bruijn):
  - Binding information only at bound occurrences (indexes).
  - At binding point, only λ to mark structure.
  - Substitution may require de Bruijn lifting of the implanted term.
- Abstraction by maps:
  - Binding information only at binding occurrences (maps).
  - At bound points, only $\square$ to mark structure.
  - No adjustment required for substitution.

Formalization

- Everything that follows is formalized in Isabelle/HOL.
  - The apparent quotients and partial functions are coded in HOL without any actual quotienting of datatypes or “domain predicates” of functions.
  - Correctness of the map representation is proved w.r.t. Nominal Isabelle.
  - Independently, correctness of the map representation is proved w.r.t. de Bruijn nameless terms in Minlog.
- However our favorite form of the map approach is not representable in HOL or easily representable in Coq:
  - Requires induction-recursion or induction-induction.

Maps, $\mathbb{M}$, defined inductively

- Maps are binary trees over 0 and 1, with the identification $(0\ 0) = 0$.
- Can formalize this inductively without quotienting using an auxiliary type $\mathbb{M}^+$ not containing 0:

$$\frac{}{1 \in \mathbb{M}^+} \qquad \frac{m^+ \in \mathbb{M}^+}{\mathsf{inl}(m^+) \in \mathbb{M}^+} \qquad \frac{n^+ \in \mathbb{M}^+}{\mathsf{inr}(n^+) \in \mathbb{M}^+}$$

$$\frac{m^+ \in \mathbb{M}^+ \qquad n^+ \in \mathbb{M}^+}{\mathsf{cons}(m^+, n^+) \in \mathbb{M}^+}$$

- Think of $\mathsf{inl}(m^+)$ as the map $(m\ 0)$.
- Extend $\mathbb{M}^+$ with 0 to get $\mathbb{M}$

$$\frac{}{0 \in \mathbb{M}} \qquad \frac{m^+ \in \mathbb{M}^+}{m^+ \in \mathbb{M}}$$

Map application

- For “cons” on $\mathbb{M}$ we define:

$$mapp(m,n) := \begin{cases} 0 & \text{if } m = n = 0, \\ \mathrm{inl}(m) & \text{if } m \neq 0 \text{ and } n = 0, \\ \mathrm{inr}(n) & \text{if } m = 0 \text{ and } n \neq 0, \\ \mathrm{cons}(m,n) & \text{if } m \neq 0 \text{ and } n \neq 0. \end{cases}$$

(Eliding explicit inclusion of $\mathbb{M}^+$ in $\mathbb{M}$.)

- Write $(m\ n)$ for $mapp(m,n)$, $(m_1 m_2\ m_3)$ for $((m_1\ m_2)\ m_3)$, etc.
- $mapp$ is injective.

Orthoganality on maps

- A symmetric *orthogonality* relation $\perp$:

$$\frac{}{m \perp 0} \qquad \frac{}{0 \perp n} \qquad \frac{m \perp n \quad m' \perp n'}{mm' \perp nn'}$$

- $m \perp n$ means:
  - m and n have the same shape
  - m and n bind different positions in that shape.
- 0 has every shape and binds no positions.
- 1 has the shape of a single hole, and binds one position.

Lambda terms as a subtype

- Symbolic expressions ($\mathbb{S}$) are raw syntax:

$$\frac{}{x \in \mathbb{S}} \qquad \frac{}{\square \in \mathbb{S}} \qquad \frac{S \in \mathbb{S} \quad T \in \mathbb{S}}{(S\ T) \in \mathbb{S}} \qquad \frac{m \in \mathbb{M} \quad S \in \mathbb{S}}{m \backslash S \in \mathbb{S}}$$

- Well formedness ($m \mid S$; m *divides* S):

$$\frac{}{0 \mid x} \qquad \frac{}{0 \mid \square} \qquad \frac{}{1 \mid \square} \qquad \frac{m \mid S \quad n \mid T}{mn \mid ST}$$

$$\frac{m \mid T \quad n \mid T \quad m \perp n}{m \mid (n \backslash T)}$$

  - $m \mid S$ means
    " S is well-formed and m is a position of unbound boxes in S ".
  - $m \mid S \Longrightarrow 0 \mid S$.
  - $0 \mid S$ means " S is well formed".

Aside: Syntax and well-formedness simultaneously

- $\mathbb{L}$ is a type.

$$\frac{}{x \in \mathbb{L}} \qquad \frac{}{\Box \in \mathbb{L}} \qquad \frac{M \in \mathbb{L} \quad N \in \mathbb{L}}{(M\ N) \in \mathbb{L}}$$

$$\frac{m \in \mathbb{M} \quad M \in \mathbb{L} \quad m \mid M}{m \backslash M \in \mathbb{L}}$$

- Divides is a relation $\mid\ \subseteq \mathbb{M} \times \mathbb{L}$.

$$\frac{}{0 \mid x} \qquad \frac{}{0 \mid \Box} \qquad \frac{}{1 \mid \Box} \qquad \frac{m \mid M \quad n \mid N}{(m\ n) \mid (M\ N)}$$

$$\frac{m \mid N \quad n \mid N \quad m \perp n}{m \mid (n \backslash N)}$$

- Not simultaneous inductive definition due to $\mathbb{L}$ in the type of $\mid$.
- Need induction-induction or induction-recursion to formalize.

Hole filling

- Define the partial operation $M_m[P] : \mathbb{L} \times \mathbb{M} \times \mathbb{L} \to \mathbb{L}$:

$$\begin{aligned}
\square_1[P] &:= P. \\
\square_0[P] &:= \square. \\
x_0[P] &:= x. \\
(M\ N)_{(m\ n)}[P] &:= (M_m[P]\ N_n[P]) && \text{if } m \mid M \text{ and } n \mid N. \\
(n \backslash N)_m[P] &:= n \backslash (N_m[P]) && \text{if } m \mid (n \backslash N).
\end{aligned}$$

- Only defined if $m \mid M$ (m is a position of unbound holes in M).
- Hole filling is a homomorphism, even going under binders.
- Hole filling respects well-formedness:

$$m \mid M \wedge 0 \mid N \Longrightarrow 0 \mid M_m[N].$$

- Why is the last equation well-formed?

Parameters: map, skeleton, abstraction

- *map*, M_x, computes the map of all the occurrences of x in M.
- *skel*, M^x, replaces all occurrences of x in M by $\square$.

$$
\begin{aligned}
\mathit{map}:\quad & \mathbb{X}\times\mathbb{L}\to\mathbb{M} \\
y_x := & \begin{cases} 1 & \text{if } x = y, \\ 0 & \text{if } x \neq y. \end{cases} \\
\square_x := & \ 0. \\
(M\ N)_x := & \ (M_x\ N_x). \\
(m\backslash M)_x := & \ M_x.
\end{aligned}
\qquad
\begin{aligned}
\mathit{skel}:\quad & \mathbb{X}\times\mathbb{L}\to\mathbb{L} \\
y^x := & \begin{cases} \square & \text{if } x = y, \\ y & \text{if } x \neq y. \end{cases} \\
\square^x := & \ \square. \\
(M\ N)^x := & \ (M^x\ N^x). \\
(m\backslash M)^x := & \ m\backslash M^x.
\end{aligned}
$$

- With map and skel can define abstraction of a name from a term.

$$\mathit{lam}(x, M) := M_x\backslash M^x$$

  - $\mathit{lam}(x, M)$ does not contain x.

Substitution defined by hole filling

$$\begin{aligned} subst: &\quad \mathbb{L} \times \mathbb{X} \times \mathbb{L} \to \mathbb{L} \\ M\{x\backslash P\} &:= (M^x)_{M_x}[P]. \end{aligned}$$

- Some provable equations of substitution

$$\begin{aligned} y\{x\backslash P\} &= \begin{cases} P & \text{if } x = y, \\ y & \text{if } x \neq y. \end{cases} \\ \square\{x\backslash P\} &= \square. \\ (M\ N)\{x\backslash P\} &= (M\{x\backslash P\}\ N\{x\backslash P\}) && \text{if } 0 \mid M \text{ and } 0 \mid N. \\ (m\backslash M)\{x\backslash P\} &= (m\backslash M\{x\backslash P\}) && \text{if } m \mid M. \end{aligned}$$

- These equations eliminate substitution on concrete terms.
- Substitution is a homomorphism.
  - There are no name-freshness conditions on these equations.

Substitution lemma of λ-calculus: better proof

If $x \neq y$ and $x \sharp P$, then

$$M\{x\backslash N\}\{y\backslash P\} = M\{y\backslash P\}\{x\backslash N\{y\backslash P\}\}.$$

- In named representations (including locally nameless and nominal) this proof requires choosing a fresh name.
    - When $M = \lambda z.M'$ we must assume $z \sharp (x, y, N, P)$
    - By equivariance, strengthened induction principle, . . .

Our proof

- By induction on (well-formedness of) M.
- Each case completely solved by equational reasoning.
    - Using the equations of substitution and the IH.
    - No need for fresh names to appy the equations of substitution. □

Datatype $\boldsymbol{\Lambda}$ of raw $\boldsymbol{\lambda}$-syntax

$$\frac{}{x \in \boldsymbol{\Lambda}} \qquad \frac{}{\Box \in \boldsymbol{\Lambda}} \qquad \frac{K \in \boldsymbol{\Lambda} \quad L \in \boldsymbol{\Lambda}}{(K\ L) \in \boldsymbol{\Lambda}} \qquad \frac{K \in \boldsymbol{\Lambda}}{\mathrm{lam}(x, K) \in \boldsymbol{\Lambda}}$$

- Define *map* (K_x) and *skel* (K^x) on $\boldsymbol{\Lambda}$
  - K_x computes the map of occurrences of x in K .
  - K^x replaces every x in K with $\Box$.

$$\begin{aligned}
\mathit{map} :\quad & \mathbb{X} \times \boldsymbol{\lambda} \to \mathbb{M} & \qquad \mathit{skel} :\quad & \mathbb{X} \times \boldsymbol{\lambda} \to \boldsymbol{\lambda} \\
y_x := & \begin{cases} 1 & \text{if } x = y, \\ 0 & \text{if } x \neq y. \end{cases} & y^x := & \begin{cases} \Box & \text{if } x = y, \\ y & \text{if } x \neq y. \end{cases} \\
\Box_x := & \ 0. & \Box^x := & \ \Box. \\
(K\ L)_x := & \ (K_x\ L_x). & (K\ L)^x := & \ (K^x\ L^x). \\
\mathrm{lam}(y, K)_x := & \begin{cases} 0 & \text{if } x = y, \\ K_x & \text{if } x \neq y. \end{cases} & \mathrm{lam}(y, K)^x := & \begin{cases} \mathrm{lam}(y, K) & \text{if } x = y, \\ \mathrm{lam}(y, K^x) & \text{if } x \neq y. \end{cases}
\end{aligned}$$

α-equivalence on Λ

- α-equivalence is defined as a relation:

$$\frac{}{x =_\alpha x} \qquad \frac{}{\Box =_\alpha \Box} \qquad \frac{K =_\alpha K' \quad L =_\alpha L'}{(K\ L) =_\alpha (K'\ L')}$$

$$\frac{K_x = L_y \quad K^x =_\alpha L^y}{\mathrm{lam}(x, K) =_\alpha \mathrm{lam}(y, L)}$$

- In the paper we prove this corresponds to a standard definition.
  - Messy proof, like any reasoning about Λ.
- $=_\alpha$ is clearly decidable.
- No fresh names or name swapping is required to decide $=_\alpha$ ...
  - ... but the proof of correctness uses fresh names and equivariance of $=_\alpha$.

Substitution on Λ, defined as a relation

$$\frac{}{x\{x\backslash J\} \to J} \qquad \frac{x \neq y}{y\{x\backslash J\} \to y} \qquad \frac{}{\Box\{x\backslash J\} \to \Box}$$

$$\frac{K\{x\backslash J\} \to L \qquad K'\{x\backslash J\} \to L'}{(K\ K')\{x\backslash J\} \to (L\ L')} \qquad \frac{z \sharp \{x, J\} \qquad K\{x\backslash J\} \to L}{\mathrm{lam}(z, K)\{x\backslash J\} \to \mathrm{lam}(z, L)}$$

$$\frac{K =_\alpha K' \qquad J =_\alpha J' \qquad K'\{x\backslash J'\} \to L' \qquad L' =_\alpha L}{K\{x\backslash J\} \to L}$$

- Correctness of substitution:
  - (Existence) $\exists L.\ K\{x\backslash J\} \to L$.
  - (Uniqueness and Congruence)

$$\frac{K\{x\backslash J\} \to L \qquad J =_\alpha J' \qquad K =_\alpha K'}{K'\{x\backslash J'\} \to L') \Leftrightarrow L =_\alpha L'}$$

- Messy proof, like any reasoning about Λ.

Relation between $\mathbb{L}$ and $\boldsymbol{\Lambda}$

- View raw terms as names for ideal terms in $\mathbb{L}$.
  - A raw term K *denotes* an ideal term $[\![K]\!]$.

$$\begin{aligned} [\![x]\!] &:= x. \\ [\![\square]\!] &:= \square. \\ [\![(K\ L)]\!] &:= ([\![K]\!]\ [\![L]\!]). \\ [\![\mathrm{lam}(x, K)]\!] &:= \mathit{lam}(x, [\![K]\!]). \end{aligned}$$

  - (Recall the definition $\mathit{lam}(x, M) := M_x \backslash M^x$.)
- Properties of denotation (have been proven directly)
  1. $M \in \mathbb{L} \implies \exists K \in \boldsymbol{\Lambda}.\ [\![K]\!] = M$. (Every term has a name.)
  2. $K =_\alpha L \iff [\![K]\!] = [\![L]\!]$.
     (α -equivalent names denote same term.)
  3. $K\{x \backslash J\} \to L \iff [\![K]\!]\{x \backslash [\![J]\!]\} = [\![L]\!]$.
     (Denotation commutes with substitution.)

Correctness of $\mathbb{L}$ w.r.t. Nominal Isabelle

- Define an inverse to $\llbracket \cdot \rrbracket$

$$\begin{aligned}
\llbracket \cdot \rrbracket &: \quad \text{Nom} \to \mathbb{L} & \lfloor \cdot \rfloor &: \quad \mathbb{L} \to \text{Nom} \\
\llbracket x \rrbracket &:= x & \lfloor x \rfloor &:= x \\
\llbracket \Box \rrbracket &:= \Box & \lfloor \Box \rfloor &:= \Box \\
\llbracket (K\ L) \rrbracket &:= (\llbracket K \rrbracket\ \llbracket L \rrbracket) & \lfloor (M_1\ M_2) \rfloor &:= (\lfloor M_1 \rfloor\ \lfloor M_2 \rfloor) \\
\llbracket \text{lam}(x, K) \rrbracket &:= lam(x, \llbracket K \rrbracket) & \lfloor m \backslash M \rfloor &:= \text{lam}(x, \lfloor M_m[x] \rfloor) \quad \text{if } x \sharp M.
\end{aligned}$$

- To get a name for $m \backslash M$, fill hole m in M with fresh parameter x , compute a name for that term, then abstract x .
- $\llbracket \cdot \rrbracket$ and $\lfloor \cdot \rfloor$ are provably functions.
  - Depends on α being identity in nominal.
- $\llbracket \cdot \rrbracket$ and $\lfloor \cdot \rfloor$ are inverses.
- $\llbracket K\{x \backslash J\} \rrbracket = \llbracket K \rrbracket \{x \backslash \llbracket J \rrbracket\}$.

$\beta\eta$-reduction

$$\frac{}{(n\backslash N)M \to_{\beta\eta} N_n[M]}\ \beta \qquad \frac{}{(01\backslash M\square) \to_{\beta\eta} M}\ \eta$$

$$\frac{M \to_{\beta\eta} M'}{MN \to_{\beta\eta} M'N}\ \text{appl} \qquad \frac{N \to_{\beta\eta} N'}{MN \to_{\beta\eta} MN'}\ \text{appr}$$

$$\frac{M \to_{\beta\eta} N}{lam(x, M) \to_{\beta\eta} lam(x, N)}\ \xi$$

- Implicitly assuming every term is well-formed.

η rule is name-free

$$\frac{}{(01\backslash M\square) \to_{\beta\eta} M}\ \eta$$

- η rule is about abstraction, not about parameters.
- Informal η-rule requires freshness condition $x \notin \mathrm{FP}(M)$.

$$\frac{x \notin \mathrm{FP}(M)}{\mathrm{lam}(x, Mx) \to_{\beta\eta} M}\ \eta$$

- Even canonical representations like de Bruijn need this condition.
- Map representation avoids this because

$$\mathrm{lam}(x, Mx) = 01\backslash M\square \qquad \text{if } x \notin \mathrm{FP}(M).$$

- (HOAS also avoids this freshness condition.)

β rule is name-free

$$\frac{}{(n\backslash N)M \to_{\beta\eta} N_n[M]}\ \beta$$

- Rule β is about abstraction and hole filling, not about parameters.
- But, β rule is name free only by accident …
  - … same abstraction on both sides of the relation.
- Rule β of parallel reduction does not have this property. Informal β-rule:

$$\frac{}{(\lambda x.\, N)\; M \to_{\beta\eta} N\{x\backslash M\}}$$

Informal parallel β-rule:

$$\frac{N \Rightarrow N' \qquad M \Rightarrow M'}{(\lambda x.\, N)\; M \Rightarrow N'\{x\backslash M'\}}$$

x is bound in N on LHS, and in N' on RHS.

ξ rule not name-free

$$\frac{M \rightarrow_{\beta\eta} N}{lam(x, M) \rightarrow_{\beta\eta} lam(x, N)}\ \xi$$

- A name bound in the conclusion is free in the premise.

$$\frac{(1\backslash\Box)x \rightarrow_{\beta\eta} x}{lam(x, (1\backslash\Box)x) \rightarrow_{\beta\eta} lam(x, x)}$$

- An incorrect name free rule ξ:

$$\frac{M \rightarrow_{\beta\eta} N}{m\backslash M \rightarrow_{\beta\eta} m\backslash N}$$

Why do we care about rules being name-free?

- Try to prove

  $$M_1 \rightarrow_{\beta\eta} M_2 \implies M_1\{x\backslash N\} \rightarrow_{\beta\eta} M_2\{x\backslash N\}$$

  by rule induction on $M_1 \rightarrow_{\beta\eta} M_2$.
  - Case for rule ξ, where $M_1 = \lambda y.\, P$:
  - Must α-convert M_1 so that $y \,\sharp\, (x, N)$, allowing substitution to go under the binder so the induction hypothesis can be used.
  - Requires equivariance and name-swapping: a well-known can of worms we wish to avoid.
- Rule ξ is not the only problematic rule.
  - E.g. β rule in parallel reduction shown above.
- An outstanding problem for map representation.

Fixing rule ξ

All of the following are correct and equivalent:

- Names in premise and conclusion:

$$\frac{M \rightarrow_{\beta\eta} N}{lam(x, M) \rightarrow_{\beta\eta} lam(x, N)}\ \xi$$

- Names only in premises, but weak induction principle:

$$\frac{x \sharp (M, N) \qquad M_m[x] \rightarrow_{\beta\eta} N_n[x]}{m \backslash M \rightarrow_{\beta\eta} n \backslash N}$$

- Names only in premises, strong induction principle:

$$\frac{\forall x \,.\, M_m[x] \rightarrow_{\beta\eta} N_n[x]}{m \backslash M \rightarrow_{\beta\eta} n \backslash N}$$

Rule induction

- We have shown an equivalent ξ rule that allows all the expected rule inductions
  - Similarly, parallel reduction can be fixed to allow expected rule inductions.
- But these fixes have all the problems we mentioned for existing approaches:
  - Equivalent systems must be defined.
  - The equivalence proofs are hard, and depend on equivariance and name swapping.
  - The equivalent systems are only expressible in a strong logic.

Can we do better?

Work in progress. (But don't hold your breath.)

- James McKinna suggested an approach that doesn't need generalized inductive definitions . . .
  - . . . but still uses alternative equivalent rules . . .
  - . . . and the proofs are even harder than existing aproaches.
- Masahiko Sato has suggested an approach that is so hard I haven't yet been able to do the proofs.

Conclusion

- A canonical presentation of λ-terms using maps.
  - Proved correct w.r.t. nominal terms (in Isabelle).
  - Proved correct w.r.t. pure de Bruijn (in Minlog, not discussed here),
  - Substitution lemma proved without renaming.
- Used maps to study raw λ syntax.
  - Decide α-conversion without renaming.
  - Substitution defined and studied.
  - Relationship with map-terms proved.
- Used maps to study de Bruijn terms (not discussed here).
- $\beta\eta$-reduction of map-terms defined.
  - Some rules are pretty in this presentation.
  - Work in progress: we do not yet have more elegant rule induction than the usual equivariance approach.