1　動機

次のような性質を持つ型理論が欲しい。

- より自然に property に関する subtyping が使える
  - 2 が自然数でもあり偶数でもある。Coq の場合は 2 と 2 が偶数であることの証明の組が偶数として型付けされる。
  - 部分集合が本当に部分集合になり、キャストが簡単（書かなくていい）
  - 結果として型付けの一意性はないと思うけど、それでもいい
- 証明項を真に区別する必要がない or 証明項を扱うことができない
  - 群が等しいとは群の演算が等しいこと、証明項まで等しいこととみなしたくない
  - 証明項を構成することもできるが、それの存在を覚えておくだけぐらいでいい
  - あと関数の外延性などの axiom をいい感じにしたい
- 構造に関する部分型（？）も使えると楽
  - 環は群の部分型とみなしたい（キャストを明示的に書きたくない）
  - これをやると部分空間の扱いが絶対にめんどくさい
  - 公称型みたいな感じで扱った方がいいかも
- 等式をもっと簡単に扱いたい、　well-definedness をもっと簡単に
  - 例として、商群からの写像の扱いが Coq ではめんどくさい
  - （部分集合系が扱えると良いなあ）

src の方については以降最新のものに従う。

2　型理論 1

ただ単に証明木 + refinement type を扱えるようにしただけの体系を考える。subtyping はない。証明項というものを完全になくそうとした場合、証明を組むのは証明項ではなく証明木を作ることになる。問題点としては、もしシステムとして実装するなら証明木を対話的に組んでいく必要があってよりめんどくさそう。なので、この方向は使わない。(うまくいかないことがあっても仕方ない。)一応書いておいただけ。

■項やコンテキストの定義

$\langle \textit{term;t,A,P} \rangle$ ::= $\langle \textit{variable;x} \rangle$
| ‘`Prop`’
| ‘`Type`’
| ‘`Fun`’ $\langle \textit{variable} \rangle$ $\langle \textit{term} \rangle$ $\langle \textit{term} \rangle$
| ‘`For`’ $\langle \textit{variable} \rangle$ $\langle \textit{term} \rangle$ $\langle \textit{term} \rangle$
| ‘`App`’ $\langle \textit{term} \rangle$ $\langle \textit{term} \rangle$
| ‘`Ref`’ $\langle \textit{term} \rangle$ $\langle \textit{term} \rangle$

$\langle \textit{context-snippet} \rangle$ ::= $\langle \textit{variable} \rangle$ ‘`:`’ $\langle \textit{term} \rangle$
| Hold $\langle \textit{term} \rangle$

$\langle \textit{context;}\Gamma \rangle$ ::= ‘`empty`’ | $\langle \textit{context} \rangle$ ‘`,`’ $\langle \textit{context-snippet} \rangle$

`Ref` が refinement type の型。コンテキストの Hold $\langle \textit{term} \rangle$ は命題の仮定を表す。t と A は同じ項だが、気持ちとしては項と型の分け方である。また `Sort` は `Prop` か `Type` を表す。自由変数とか subst とかはめんどくさいので書いてない。 conversion rule も定義していないが、 $M_1 \equiv_\beta M_2$ みたいなのが定義されていてほしい。

コンテキストと項の上の関係 (rule) を定める。直観的にはコンテキストの well-def 性、型付け可能性、証明可能、を表す。

rule 一覧

$\vdash \Gamma$
$\Gamma \vdash t_1 : t_2$
$\Gamma \vDash t$

「コンテキストに含まれる自由変数」などは定義してないが、いい感じに定義されているとする。ここでは β 同値は rule に含まれない。これらの rule に関して以下のように rule の間の関係を定める。(ここで rule と異なるものが並んでいる場合は、ここではそれを条件とでも呼ぶこともある。例えば、 β 同値は条件である。ただし標準的な呼び方かはわからない。)(Type : Type を避けるためにいろいろ書いているが無駄だったかもしれない。)

コンテキストの well-def

$$\frac{}{\vdash \mathit{empty}}\ \text{context empty}$$

$$\frac{\vdash \Gamma}{\vdash \Gamma, x : \mathtt{Type}}\ \text{context type}$$

$$\frac{\vdash \Gamma \quad \Gamma \vdash A : \mathtt{Type} \quad x \notin \Gamma}{\vdash \Gamma, x : A}\ \text{context term}$$

$$\frac{\vdash \Gamma \quad \Gamma \vdash P : \mathtt{Prop}}{\vdash \Gamma, \mathrm{Hold}\, P}\ \text{context prop}$$

コンテキストから自明

$$\frac{\vdash \Gamma \quad \Gamma \vdash A : \mathtt{Type}}{\Gamma, x : A \vdash x : A}\ \text{start}$$

$$\frac{\vdash \Gamma, x : A_1 \quad \Gamma \vdash t : A_2}{\Gamma, x : A_1 \vdash t : A_2}\ \text{weakning}$$

型付け (form)

$$\frac{\Gamma \vdash A_1 : \mathtt{Type} \quad \Gamma, x : A_1 \vdash A_2 : \mathtt{Type}}{\Gamma \vdash \mathtt{For}\, x\, A_1\, A_2 : \mathtt{Type}}\ \text{forall formation(type)}$$

$$\frac{\Gamma \vdash A_1 : \mathtt{Type} \quad \Gamma, x : A_1 \vdash A_2 : \mathtt{Prop}}{\Gamma \vdash \mathtt{For}\, x\, A_1\, A_2 : \mathtt{Prop}}\ \text{forall formation(prop)}$$

$$\frac{\Gamma \vdash A_1 : \mathtt{Type} \quad \Gamma, x : A_1 \vdash A_2 : \mathtt{Prop}}{\Gamma \vdash \mathtt{For}\, x\, A_1\, A_2 : \mathtt{Prop}}\ \text{forall formation(prop2)}$$

$$\frac{\Gamma \vdash A : \mathtt{Type} \quad \Gamma \vdash P : \mathtt{For}\, x\, A\, \mathtt{Prop}}{\Gamma \vdash \mathtt{Ref}\, A\, P : \mathtt{Type}}\ \text{refinement formation}$$

Prop から Type を作るのは禁止したいので型付けられないようにしてある

conversion

$$\frac{\Gamma \vdash x : A_1 \quad A_1 \equiv_\beta A_2 \quad \Gamma \vdash A_2 : \mathtt{Type}}{\Gamma \vdash x : A_2}\ \text{conversion(type)}$$

$$\frac{\Gamma \vdash A_1 : \mathtt{Type} \quad A_1 \equiv_\beta A_2}{\Gamma \vdash A_2 : \mathtt{Type}}\ \text{(type)}$$

$$\frac{\Gamma \vdash A_1 : \mathtt{Prop} \quad A_1 \equiv_\beta A_2}{\Gamma \vdash A_2 : \mathtt{Prop}}\ \text{(prop)}$$

application や conversion により 下二つはいらないかもしれないし、逆に強すぎるかもしれない。(subject reduction (?) がうまくいくのかわからなかったのでとりあえずつけた。 ITT での equality に関する judgement みたいなのがないのでバグがありそう）変な問題があっても直す気はない。

型付け (intro と elim)

$$\frac{\Gamma, x : A_1 \vdash t : A_2 \quad \Gamma \vdash \texttt{For}\, x\, A_1\, A_2 : \texttt{Type}}{\Gamma \vdash \texttt{Fun}\, x\, A_1\, t : \texttt{For}\, x\, A_1\, A_2} \text{ forall introduction}$$

$$\frac{\Gamma \vdash t_1 : \texttt{For}\, x\, A_1\, A_2 \quad \Gamma \vdash t_2 : A_1}{\Gamma \vdash \texttt{App}\, t_1\, t_2 : A_2\{x \leftarrow t_2\}} \text{ forall elimination}$$

$$\frac{\Gamma \vdash t : A \quad \Gamma \vdash \texttt{Ref}\, A\, P : \texttt{Type} \quad \Gamma \vDash \texttt{App}\, P\, t}{\Gamma \vdash t : \texttt{Ref}\, A\, P} \text{ refinement introduction}$$

$$\frac{\Gamma \vdash t : \texttt{Ref}\, A\, P}{\Gamma \vdash t : A} \text{ refinement elimination}$$

Proof

$$\frac{\Gamma \vdash P_1 : \texttt{Prop} \quad P_1 \equiv_\beta P_2 \quad \Gamma \vDash P_1}{\Gamma \vDash P_2} \text{ prop conversion}$$

$$\frac{\Gamma \vdash \texttt{For}\, x\, A\, P : \texttt{Prop} \quad \Gamma, x : A \vDash \texttt{App}\, P\, x}{\Gamma \vDash \texttt{For}\ x\, A\, P} \text{ forall intro(prop)}$$

$$\frac{\Gamma \vDash \texttt{For}\, x\, A\, P \quad \Gamma \vdash t : A}{\Gamma \vDash P\{x \leftarrow t\}} \text{ forall elim(prop)}$$

$$\frac{\Gamma \vdash x : \texttt{Ref}\, A\, P}{\Gamma \vDash \texttt{App}\, P\, x} \text{ refinement elim(prop)}$$

これによくある論理の何某を付け加えて終わり。この型理論を検討することはないと思うが、例を後で挙げたい。Prop と Type で別々の型付けが必要になることがありめんどくさい。

3　型理論 2

証明項に対応するものを微妙に導入して、型システムが自動的に判定できる証明を増やした形。

■項やコンテキストの定義

$\langle \mathit{term} \rangle$ ::= $\langle \mathit{variable} \rangle$
| ‘`Prop`’
| ‘`Type`’
| ‘`Fun`’ $\langle \mathit{variable} \rangle$ $\langle \mathit{term} \rangle$ $\langle \mathit{term} \rangle$
| ‘`For`’ $\langle \mathit{variable} \rangle$ $\langle \mathit{term} \rangle$ $\langle \mathit{term} \rangle$
| ‘`App`’ $\langle \mathit{term} \rangle$ $\langle \mathit{term} \rangle$
| ‘`Ref`’ $\langle \mathit{term} \rangle$ $\langle \mathit{term} \rangle$
| ‘`Prf`’ $\langle \mathit{term} \rangle$

$\langle \mathit{context\text{-}snippet} \rangle$ ::= $\langle \mathit{variable} \rangle$ ‘`:`’ $\langle \mathit{term} \rangle$
| Hold $\langle \mathit{term} \rangle$

$\langle \mathit{context} \rangle$ ::= ‘`empty`’ | $\langle \mathit{context} \rangle$ ‘`,`’ $\langle \mathit{context\text{-}snippet} \rangle$

`Proof` は証明項を陽に扱うためのもの。命題 P が示せるときに `Proof` P を証明項として扱ってよいとすることで簡単にならないだろうか？あるいは、（証明が必要となるところでのみ用いたい）

■項やコンテキストの評価　ここでコンテキストや項の関係を定義していく。

コンテキストと項の関係

$\vdash \Gamma$, コンテキストの well-def 性
$\Gamma \vdash t_1 : t_2$, 項の型付け性
$\Gamma \vDash t$, 項の証明可能性

コンテキストの well-def

$$\frac{}{\vdash \mathit{empty}}\ \text{context empty}$$

$$\frac{\vdash \Gamma \quad \Gamma \vdash A : \mathtt{Sort} \quad x \notin \mathrm{FV}(\Gamma)}{\vdash \Gamma, x : A}\ \text{context start}$$

$$\frac{\vdash \Gamma \quad \Gamma \vdash P : \mathtt{Prop}}{\vdash \Gamma, \mathrm{Hold}\ P}\ \text{context prop}$$

コンテキスト 自明

$$\frac{}{\mathit{empty} \vdash \mathtt{Sort}_1 : \mathtt{Sort}_2}\ \text{axiom}$$

$$\frac{\vdash \Gamma \quad \Gamma \vdash A : \mathtt{Type} \quad x \notin \mathrm{FV}(\Gamma)}{\Gamma, x : A \vdash x : A}\ \text{variable}$$

$$\frac{\vdash \Gamma, \_ \quad \Gamma \vdash t : A_2}{\Gamma, \_ \vdash t : A_2}\ \text{weakning}$$

formation

$$\frac{\Gamma \vdash A_1 : \mathtt{Sort}_1 \quad \Gamma, x : A_1 \vdash A_2 : \mathtt{Sort}_2}{\Gamma \vdash \mathtt{For}\, x\, A_1\, A_2 : \mathtt{Sort}_2}\ \text{forall formation}$$

$$\frac{\Gamma \vdash A : \mathtt{Type} \quad \Gamma \vdash P : \mathtt{For}\, x\, A\, \mathtt{Prop}}{\Gamma \vdash \mathtt{Ref}\, A\, P : \mathtt{Type}}\ \text{refinement formation}$$

β 同値について

$$\frac{\Gamma \vdash x : A_1 \quad A_1 \equiv_\beta A_2 \quad \Gamma \vdash A_2 : \mathtt{Sort}}{\Gamma \vdash x : A_2}\ \text{conversion}$$

introduction と elimination

$$\frac{\Gamma, x : A_1 \vdash t : A_2 \quad \Gamma \vdash \mathtt{For}\, x\, A_1\, A_2 : \mathtt{Type}}{\Gamma \vdash \mathtt{Fun}\, x\, A_1\, t : \mathtt{For}\, x\, A_1\, A_2}\ \text{for intro} \quad \frac{\Gamma \vdash t_1 : \mathtt{For}\, x\, A_1\, A_2 \quad \Gamma \vdash t_2 : A_1}{\Gamma \vdash \mathtt{App}\, t_1\, t_2 : A_2\{x \leftarrow t_2\}}\ \text{for elim}$$

$$\frac{\Gamma \vdash t : A \quad \Gamma \vdash \mathtt{Ref}\, A\, P : \mathtt{Type} \quad \Gamma \vDash \mathtt{App}\, P\, t}{\Gamma \vdash t : \mathtt{Ref}\, A\, P}\ \text{ref intro} \quad \frac{\Gamma \vdash t : \mathtt{Ref}\, A\, P}{\Gamma \vdash t : A}\ \text{ref elim}$$

proof term について

$$\frac{}{\Gamma, \mathrm{Hold}\, P \vDash P}\ \text{assumption} \quad \frac{\Gamma \vDash P}{\Gamma, \_ \vDash P}\ \text{weakning}$$

$$\frac{\Gamma \vdash P : \mathtt{Prop} \quad \Gamma \vdash t : P}{\Gamma \vDash P}\ \text{implicit proof} \quad \frac{\Gamma \vdash P : \mathtt{Prop} \quad \Gamma \vDash P}{\Gamma \vdash \mathtt{Prf}\, P : P}\ \text{explicit proof}$$

$$\frac{\Gamma \vdash t : \mathtt{Ref}\, A\, P}{\Gamma \vDash \mathtt{App}\, P\, t}\ \text{refinement inversion}$$

具体例を出したい。

- $A : \texttt{Prop}, B : \texttt{Prop}, \mathrm{Hold}\,(\texttt{for}\,x\,A\,B), \mathrm{Hold}\,A \vDash B$
- $A, B : \texttt{Type}$, $f : A \to B$ とする。$P, Q : A, B$ 上の述語に対して 「任意の $x : A$ について、 $P(x)$ が成り立つなら $Q(f(x))$ が成り立つ」とする。このとき $f : \texttt{Ref}\,A\,P \to \texttt{Ref}B\,Q$ と型付けられるはず。

一つ目は省略（カリーハワードと Proof を使う）コンテキストとしては、

$$\Gamma := A : \texttt{Type}, B : \texttt{Type}, f : A \to B, P : A \to \texttt{Prop}, Q : B \to \texttt{Prop}$$

をまず考え、これに命題に対応する項がくっつく。$K := \texttt{For}\,x\,A\,((\texttt{App}\,P\,x) \to (\texttt{App}\,Q\,(\texttt{App}\,f\,x))))$ なる $\texttt{Prop}$ 型の項が 「任意の $x : A$ について、 $P(x)$ が成り立つなら ...」に対応する。改めて $\Gamma \leftarrow \Gamma, \mathrm{Hold}\ K$ と置き直して、 $\Gamma \vdash f : \texttt{Ref}\,A\,P \to \texttt{Ref}\,B\,Q$ を示す。eta-conversion が必要になっちゃった。

$$\dfrac{\dfrac{\dfrac{\dfrac{\Gamma_1 \vdash f : A \to B \quad \Gamma_1 \vdash x : A}{\Gamma_1 \vdash \texttt{App}\,f\,x : B}\,\text{for intro} \quad \dfrac{\vdots}{\Gamma_1 \vdash \texttt{Ref}\,B\,Q : \texttt{Type}}\,\textit{hello} \quad \dfrac{\vdots}{\Gamma_1 \vDash \texttt{App}\,Q\,(\texttt{App}\,f\,x)}\,X_1}{\Gamma_1 := \Gamma, x : \texttt{Ref}\,A\,P \vdash \texttt{App}\,f\,x : \texttt{Ref}\,B\,Q}}{\Gamma \vdash \texttt{Fun}\,x\,(\texttt{App}\,f\,x) : \texttt{Ref}\,A\,P \to \texttt{Ref}\,B\,Q}\,\text{for elim}}{\Gamma \vdash f : \texttt{Ref}\,A\,P \to \texttt{Ref}\,B\,Q}\,\text{eta-conversion}$$

X_1 は app 省略して

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma_1 \vDash K}\,\text{assumption}}{\Gamma_1 \vdash \texttt{Prf}\,K : K}\,\text{exp prf}}{\Gamma_1 \vdash ((\texttt{Prf}\,K)\,x) : (P\,x) \to (Q\,(f\,x))}\,\text{for elim} \quad \dfrac{\dfrac{\dfrac{}{\Gamma_1 \vdash x : \texttt{Ref}\,A\,P}\,\text{var term}}{\Gamma_1 \vDash P\,x}\,\text{ref inv}}{\Gamma_1 \vdash \texttt{Prf}\,(P\,x) : P\,x}\,\text{exp prf}}{\dfrac{\Gamma_1 \vdash ((\texttt{Prf}\,K)\,x)\,(\texttt{Prf}\,(P\,x)) : \texttt{App}\,Q\,(\texttt{App}\,f\,x)}{\Gamma_1 \vDash \texttt{App}\,Q\,(\texttt{App}\,f\,x)}\,\text{imp prf}}\,\text{for elim}$$

_ ⊨ _ が必要になったときのほとんどは自動的に推論できそう

上の奴は実装するのが難しい。実装上はコンテキストの α 同値を考えたり、β 同値の判定が停止する形で扱えると嬉しい。のでもう少し扱いやすくすることにした。（同じ体系として考えられるかはわからない）$\equiv$: Term $\to$ Term $\to$ `Bool` は通常の α 同値で定義する。$\equiv$: Context $\to$ Context $\to$ `Bool` を型の部分の α 同値を許すものとして定義する。FV : Context $\to$ Variable list として $x \in \mathrm{FV}(\Gamma)$ かどうかも Bool 値で定まる。また、 1 step の β reduction （停止する）step : Term $\to$ Term が定まっているとする。特に、$M \to_\beta N$ かどうかが（停止する形）で判定できる。

β 同値性については β 同値であることの証明を受け取りそれが正しいか判定する関数を作りたい。一応 β 同値性を定義する。以下の同値閉包として定義（ただし後で作ったものがこれと一致するかは確かめてない）。

beta equivalence

$$\text{alpha} \quad \frac{}{t_1 \equiv_\beta t_2} \quad t_1 \equiv t_2$$

$$\text{step} \quad \frac{}{t_1 \equiv_\beta t_2} \quad \mathrm{step}\, t_1 \equiv t_2$$

ここで list (Term * Term) を β 同値宣言列 (betaEqs とおく。メタ変数では L) ということにする。同値宣言列の最初と最後が (Option 型に）定まる。つまり、 Begin , End : (Term * Term) list $\to$ Option Term を作る。このリストの元 (t_1, t_2) 各々が $t_1 \equiv t_2$ か $\mathrm{step}\, t_1 \equiv t_2$ か $\mathrm{step}\, t_2 \equiv t_1$ を満たすとする。このとき、宣言列は β 同値を表していると思われるから、acceptable = acc : betaEqs $\to$ Bool で β 同値性を表すものを定めることができる。

コンテキストと項の関係

$\vdash \Gamma$, コンテキストの well-def 性

$\Gamma \vdash t_1 : t_2$, 項の型付け性

$\Gamma \vDash t$, 項の証明可能性

この関係自体にも同値関係を自然に定めておく。次に規則自体を上から下に計算できるようにしておく（規則にも名前を付ける)。規則と judgement のリストをとり judgement 規則が導けるかを計算する（ option 型？)。つまり、 F : rule $\to$ Context list $\to$ Option Context であって、下に定めるものを成り立たせる関数が作れるように規則の方を変形した。その結果、コンテキストと項の関係に新しい変数の宣言が必要になった。右に成り立つ条件を書いたが、これは規則の上の条件から停止する形で判定できる。同値宣言列とか変数宣言があるのきもいわ。なくします。

コンテキストの well-def

$$\text{context empty} \quad \frac{}{\vdash \mathit{empty}}$$

$$\text{context start}\,(x) \quad \frac{\vdash \Gamma_1 \quad \Gamma_2 \vdash A : \mathtt{Type}}{\vdash \Gamma_1, x : A} \quad \Gamma_1 \equiv \Gamma_2, x \notin \mathrm{FV}(\Gamma_1)$$

$$\text{context prop} \quad \frac{\vdash \Gamma_1 \quad \Gamma_2 \vdash P : \mathtt{Prop}}{\vdash \Gamma_1, \mathrm{Hold}\ P} \quad \Gamma_1 \equiv \Gamma_2$$

コンテキスト 自明

axiom(Sort of s_1, Sort of s_2) $$\frac{}{\mathit{empty} \vdash s_1 : s_2}$$

variable (x) $$\frac{\vdash \Gamma_1 \quad \Gamma_2 \vdash A : \mathtt{Type}}{\Gamma_1, x : A \vdash x : A}$$ $\Gamma_1 \equiv \Gamma_2, x \notin \mathrm{FV}(\Gamma)$

weakning $$\frac{\vdash \Gamma_1, \_ \quad \Gamma_2 \vdash t : A_2}{\Gamma_1, \_ \vdash t : A_2}$$ $\Gamma_1 \equiv \Gamma_2$

formation

forall formation $$\frac{\Gamma_1 \vdash A_1 : \mathtt{Sort}_1 \quad \Gamma_2, x : A_2 \vdash A_3 : \mathtt{Sort}_2}{\Gamma_1 \vdash \mathtt{For}\, x\, A_2\, A_3 : \mathtt{Sort}_2}$$ $\Gamma_1 \equiv \Gamma_2, A_1 \equiv A_2$

refinement formation $$\frac{\Gamma_1 \vdash A_1 : \mathtt{Type} \quad \Gamma_2 \vdash P : \mathtt{For}\, x\, A_2\, \mathtt{Prop}}{\Gamma_1 \vdash \mathtt{Ref}\, A_1\, P : \mathtt{Type}}$$ $\Gamma_1 \equiv \Gamma_2, A_1 \equiv A_2$

conversion

conversion (L) $$\frac{\Gamma_1 \vdash t : A_1 \quad \Gamma_2 \vdash A_2 : \mathtt{Sort}}{\Gamma_1 \vdash t : A_2}$$ $\Gamma_1 \equiv \Gamma_2, \mathrm{begin}\, L \equiv A_1, \mathrm{end}\, L \equiv A_2, \mathrm{acc}\, L$

introduction と elimination

for intro $$\frac{\Gamma_1, x_1 : A_1 \vdash t : A_2 \quad \Gamma_2 \vdash \mathtt{For}\, x_2\, A_3\, A_4 : \mathtt{Type}}{\Gamma_1 \vdash \mathtt{Fun}\, x_1\, A_1\, t : \mathtt{For}\, x_1\, A_1\, A_2}$$ $\Gamma_1 \equiv \Gamma_2, A_1 \equiv A_3, A_2 \equiv A_4, x_1 = x_2$

for elim $$\frac{\Gamma_1 \vdash t_1 : \mathtt{For}\, x\, A_1\, A_2 \quad \Gamma_2 \vdash t_2 : A_3}{\Gamma \vdash \mathtt{App}\, t_1\, t_2 : A_2\{x \leftarrow t_2\}}$$ $A_1 \equiv A_3$

ref intro $$\frac{\Gamma_1 \vdash t_1 : A_1 \quad \Gamma \vdash \mathtt{Ref}\, A_2\, P_1 : \mathtt{Type} \quad \Gamma \vDash \mathtt{App}\, P_2\, t_2}{\Gamma \vdash t : \mathtt{Ref}\, A\, P}$$ $A_1 \equiv A_2, t_1 \equiv t_2, P_1 \equiv P_2$

ref elim $$\frac{\Gamma \vdash t : \mathtt{Ref}\, A\, P}{\Gamma \vdash t : A}$$

proof term について

assumption $$\frac{\vdash \Gamma, \mathrm{Hold} P}{\Gamma, \mathrm{Hold}\, P \vDash P}$$

weakning $$\frac{\vdash \Gamma, \_ \quad \Gamma \vDash P}{\Gamma, \_ \vDash P}$$

implicit proof $$\frac{\Gamma_1 \vdash P_1 : \mathtt{Prop} \quad \Gamma_2 \vdash t : P_2}{\Gamma_1 \vDash P_1}$$ $\Gamma_1 \equiv \Gamma_2, P_1 \equiv P_2$

explicit proof $$\frac{\Gamma_1 \vdash P_1 : \mathtt{Prop} \quad \Gamma_2 \vDash P_2}{\Gamma_1 \vdash \mathtt{Prf}\, P_1 : P_1}$$ $\Gamma_1 \equiv \Gamma_2, P_1 \equiv P_2$

refinement inversion $$\frac{\Gamma \vdash t : \mathtt{Ref}\, A\, P}{\Gamma \vDash \mathtt{App}\, P\, t}$$

4 型理論３

もうちょっと β 同値を明示的に扱った方が、eta-conversion とかつけやすい気がした。また、(definitional な？) functional extensionality を最終的に扱うにあたり、何か制限を書けた方がいいかも。例えば、 f_1, f_2 が共に $A \to B, \texttt{Ref}\, A\, P \to \texttt{Ref}\, B\, Q$ と型付けされるとすると、同値であることを $\texttt{Ref}\, A\, P \to \texttt{Ref}\, B\, Q$ の中で示したら、 $A \to B$ の中で同値になってしまわないように気を付ける必要がある。（あるいは $\texttt{Ref}\, A\, P \to \texttt{Ref}\, B\, Q$ での同値が $A \to B$ での同値としてもよいレベルの制限？）今回は β 同値と η 同値を見たことある形で適当に rule に入れることにした。これは rule にしなくても $\equiv$ という関係を別に定義して条件として入れてしまってもよい。あといるのかどうかよくわからなかったので、 Context Start と Axiom を制限することにした。実装してから困ったら変える。

また、 `Hold` に対応するコンストラクタとして `If` を導入する方がやりやすくなりそう。

結果として、 equivalenve を明示的に書いただけになってしまった。あとで well-defined な構成を行うときに使うと思う。例えば、 $\texttt{Take}\, x\, A\, t$ を A が inhabitants で A の取り方によらないことが示せたら、 t の型 T として、 $\texttt{Take}\, x\, A\, t$ を T に型付けられるようにしたら、商集合からの写像がうまくいくと思う。

■項やコンテキストの定義

$\langle \textit{term} \rangle$ ::= $\langle \textit{variable} \rangle$
| ‘`Prop`’
| ‘`Type`’
| ‘`Fun`’ $\langle \textit{variable} \rangle$ $\langle \textit{term} \rangle$ $\langle \textit{term} \rangle$
| ‘`For`’ $\langle \textit{variable} \rangle$ $\langle \textit{term} \rangle$ $\langle \textit{term} \rangle$
| ‘`App`’ $\langle \textit{term} \rangle$ $\langle \textit{term} \rangle$
| ‘`Ref`’ $\langle \textit{term} \rangle$ $\langle \textit{term} \rangle$
| ‘`If`’ $\langle \textit{term} \rangle$ $\langle \textit{term} \rangle$
| ‘`Prf`’ $\langle \textit{term} \rangle$

$\langle \textit{context-snippet} \rangle$ ::= $\langle \textit{variable} \rangle$ ‘`:`’ $\langle \textit{term} \rangle$
| Hold $\langle \textit{term} \rangle$

$\langle \textit{context} \rangle$ ::= ‘`empty`’ | $\langle \textit{context} \rangle$ ‘`,`’ $\langle \textit{context-snippet} \rangle$

■項やコンテキストの評価　ここでコンテキストや項の関係を定義していく。新しく項の同一性を rule に含める。そのため、木の種類が増える。

コンテキストと項の関係

$\vdash \Gamma$, コンテキストの well-def 性
$\Gamma \vdash t_1 : t_2$, 項の型付け性
$\Gamma \vDash t$, 項の証明可能性
$t_1 \equiv t_2$, 項の同一性

コンテキストの well-def

$$\frac{}{\vdash \textit{empty}}\ \text{context empty}$$

$$\frac{\vdash \Gamma \quad \Gamma \vdash A : \texttt{Type} \quad x \notin \Gamma}{\vdash \Gamma, x : A}\ \text{context start}$$

$$\frac{\vdash \Gamma \quad \Gamma \vdash P : \texttt{Prop}}{\vdash \Gamma, \mathrm{Hold}\, P}\ \text{context prop}$$

コンテキスト 自明

$$\frac{}{\textit{empty} \vdash \texttt{Sort} : \texttt{Type}}\ \text{axiom Type}$$

$$\frac{}{\textit{empty} \vdash \texttt{Prop} : \texttt{Prop}}\ \text{axiom Prop}$$

$$\frac{\vdash \Gamma \quad \Gamma \vdash A : \texttt{Type} \quad x \notin \Gamma}{\Gamma, x : A \vdash x : A}\ \text{variable}$$

$$\frac{\vdash \Gamma, \_ \quad \Gamma \vdash t : A_2}{\Gamma, \_ \vdash t : A_2}\ \text{weakning}$$

formation

$$\frac{\Gamma \vdash A_1 : \texttt{Sort}_1 \quad \Gamma, x : A_1 \vdash A_2 : \texttt{Sort}_2}{\Gamma \vdash \texttt{For}\, x\, A_1\, A_2 : \texttt{Sort}_2}\ \text{forall formation}$$

$$\frac{\Gamma \vdash A : \texttt{Type} \quad \Gamma \vdash P : \texttt{For}\, x\, A\, \texttt{Prop}}{\Gamma \vdash \texttt{Ref}\, A\, P : \texttt{Type}}\ \text{refinement formation}$$

type の conversion

$$\frac{\Gamma \vdash x : A_1 \quad A_1 \equiv A_2 \quad \Gamma \vdash A_2 : \texttt{Sort}}{\Gamma \vdash x : A_2}\ \text{type conversion}$$

introduction と elimination

$$\frac{\Gamma \vdash \texttt{For}\, x\, A_1\, A_2 : \texttt{Type} \quad \Gamma, x : A_1 \vdash t : A_2}{\Gamma \vdash \texttt{Fun}\, x\, A_1\, t : \texttt{For}\, x\, A_1\, A_2}\ \text{for intro}$$

$$\frac{\Gamma \vdash t_1 : \texttt{For}\, x\, A_1\, A_2 \quad \Gamma \vdash t_2 : A_1}{\Gamma \vdash \texttt{App}\, t_1\, t_2 : A_2\{x \leftarrow t_2\}}\ \text{for elim}$$

$$\frac{\Gamma \vdash t : A \quad \Gamma \vdash \texttt{Ref}\, A\, P : \texttt{Type} \quad \Gamma \vDash \texttt{App}\, P\, t}{\Gamma \vdash t : \texttt{Ref}\, A\, P}\ \text{ref intro}$$

$$\frac{\Gamma \vdash t : \texttt{Ref}\, A\, P}{\Gamma \vdash t : A}\ \text{ref elim}$$

$$\frac{\Gamma \vdash \texttt{For}\, x\, P_1\, P_2 \quad \Gamma, \mathrm{Hold}\, P_1 \vdash p_2 : P_2}{\Gamma \vdash \texttt{If}\, P_1\, p_2 : \texttt{For}\, x\, P_1\, P_2}\ \text{intro}$$

equiv rel

$$\frac{t_1 \equiv_\alpha t_2}{t_1 \equiv t_2}\ \text{alpha-refl}$$

$$\frac{t_2 \equiv t_1}{t_1 \equiv t_2}\ \text{sym}$$

$$\frac{t_1 \equiv t_2 \quad t_2 \equiv t_3}{t_1 \equiv t_3}\ \text{trans}$$

conversion

$$\frac{A_1^1 \equiv A_1^2 \quad A_2^1 \equiv A_2^2}{\texttt{for}\, x\, A_1^1\, A_2^1 \equiv \texttt{for}\, x\, A_1^2\, A_2^2}\ \text{for conversion}$$

$$\frac{t_1^1 \equiv t_1^2 \quad t_2^1 \equiv t_2^2}{\texttt{fun}\, x\, t_1^1\, t_2^1 \equiv \texttt{fun} x\, t_1^2\, t_2^2}\ \text{fun conversion}$$

$$\frac{t_1^1 \equiv t_1^2 \quad t_2^1 \equiv t_2^2}{\texttt{app}\, t_1^1\, t_2^1 \equiv \texttt{app}\, t_1^2\, t_2^2}\ \text{app conversion}$$

$$\frac{A_1^1 \equiv A_1^2 \quad A_2^1 \equiv A_2^2}{\texttt{ref}\, A_1^1\, A_2^1 \equiv \texttt{ref}\, A_1^2\, A_2^2}\ \text{ref conversion}$$

computation と eta

$$\frac{}{\Gamma \vdash \texttt{app}\, (\texttt{fun}\, x\, A\, t_1) t_2 \equiv t_1\{x \leftarrow t_2\}}\ \text{for computation}$$

$$\frac{}{\Gamma \vdash \texttt{fun}\, x\, A\, (\texttt{app}\, f\, x) \equiv f}\ \text{for eta}$$

proof term について

$$\frac{\vdash \Gamma, \mathrm{Hold}\, P}{\Gamma, \mathrm{Hold}\, P \vDash P}\ \text{assumption}$$

$$\frac{\Gamma \vDash P}{\Gamma, \_ \vDash P}\ \text{weakning}$$

$$\frac{\Gamma \vdash P : \texttt{Prop} \quad \Gamma \vdash t : P}{\Gamma \vDash P}\ \text{implicit proof}$$

$$\frac{\Gamma \vDash P}{\Gamma \vdash \texttt{Prf}\, P : P}\ \text{explicit proof}$$

$$\frac{\Gamma \vdash t : \texttt{Ref}\, A\, P}{\Gamma \vDash \texttt{App}\, P\, t}\ \text{refinement inversion}$$