Aside from Dave: I arrived to class just before Derek proved canonical forms for pairs. Derek later told me what he had covered and I reconstructed the first few topics. Please check the video: I am sure Derek motivated those topics better than me and his proofs may be easier to follow.

— Adequacy of the LR for open terms

We're not done proving the FTLR. Two things remain. First, we must show that contextual equivalence is well-defined; that is, there exists a largest type-respecting binary relation that is a congruence and adequate. Derek might eventually simplify and present Pitts' proof. For the moment, please refer to the “finicky” proof in ATTPL and consider ≡ctx well-defined. Second, we must show that our logical relation on open terms is adequate.

Lemma (Determinacy):

If $e \mapsto^* v_1$
and $e \mapsto^* v_2$,
then $v_1 = v_2$.

Proof: Immediate: The rules for $\mapsto$ are deterministic.

Proposition:

$\approx$ is adequate.

Proof:

WK: $\vdash e_1 \approx e_2 : \text{bool}$
$\Leftrightarrow$ $\vdash e_1 : \text{bool} \wedge \vdash e_2 : \text{bool} \wedge (e_1,e_2) \in E[\text{bool}]$
$\Leftrightarrow$ $\vdash e_1 : \text{bool} \wedge \vdash e_2 : \text{bool} \wedge$
$\exists v_1,v_2.\ e_1{\downarrow}v_1 \wedge e_2{\downarrow}v_2 \wedge (v_1,v_2) \in V[\text{bool}]$.
TS: $v_1 \equiv\text{KL}\ v_2$
$\Leftrightarrow$ $(v_1{\downarrow}\top \Leftrightarrow v_2{\downarrow}\top)$.

The proof is symmetric. We prove only one direction.
WK: v_1 [bool] true false $\downarrow$ true.
TS: v_2 [bool] true false $\downarrow$ true.

Set $R \in \text{Cand} := \{ (\text{true},\text{true}), (\text{false},\text{false}) \}$.
WK: $(v_1,v_2) \in V[\text{bool}]$
$\Rightarrow$ $(v_1\ \text{bool}, v_2\ \text{bool}) \in E[\alpha\to\alpha\to\alpha]\alpha\mapsto R$
$\Rightarrow$ $(v_1\ \text{bool true}, v_2\ \text{bool true}) \in E[\alpha\to\alpha]\alpha\mapsto R$
$\Rightarrow$ $(v_1\ \text{bool true false}, v_2\ \text{bool true false}) \in E[\alpha]\alpha\to R$
$\Leftrightarrow$ $\exists v'_1,v'_2.$
v_1 bool true false $\downarrow v'_1 \wedge$
v_2 bool true false $\downarrow v'_2 \wedge$

$(v'_1,v'_2) \in V[\alpha]\alpha\mapsto R = R.$

Since evaluation is deterministic, we have v'_1 = true.

By the definition of R, we have v'_2 = true.

Q.E.D.

— Convenient lemmas and notation

We want to use the LR to prove some theorems. In those proofs, we want to mitigate the tedium of bouncing between V and E. The following lemmas and notation help.

Lemma (Term applications):

$(e_1,e_2) \in E[\sigma\longrightarrow\tau]\rho$ iff $e_1\downarrow \wedge e_2\downarrow \wedge$

(†) $\forall(e'_1,e'_2) \in E[\sigma]\rho.\ (e_1\ e'_1,\ e_2\ e'_2) \in E[\tau]\rho.$

Proof:

($\Rightarrow$) WK: $\exists v_1,v_2.\ e_1\downarrow v_1 \wedge e_2\downarrow v_2 \wedge (v_1,v_2) \in V[\sigma\longrightarrow\tau]\rho.$

Thus, $e_1\downarrow$ and $e_2\downarrow$.

TS: $\forall(e'_1,e'_2) \in E[\sigma]\rho.\ (e_1\ e'_1,\ e_2\ e'_2) \in E[\tau]\rho.$

WK: $\exists v'_1,v'_2.\ e'_1\downarrow v'_1 \wedge e'_2\downarrow v'_2 \wedge (v'_1,v'_2) \in V[\sigma]\rho.$

WK: $(v_1,v_2) \in V[\sigma\longrightarrow\tau]\rho \wedge (v'_1,v'_2) \in V[\sigma]\rho$

$\Rightarrow$ $\exists v''_1,v''_2.\ (v_1\ v'_1)\downarrow v''_1 \wedge (v_2\ v'_2)\downarrow v''_2 \wedge (v''_1,v''_2) \in V[\tau]\rho.$

WK: $e_1\ e'_1 \mapsto^* v_1\ e'_1 \mapsto^* v_1\ v'_1 \mapsto^* v''_1 \wedge$

$e_2\ e'_2 \mapsto^* v_2\ e'_2 \mapsto^* v_2\ v'_2 \mapsto^* v''_2 \wedge$

$(v''_1,v''_2) \in V[\tau]\rho.$

($\Leftarrow$) WK: $\exists v_1,v_2.\ e_1\downarrow v_1 \wedge e_2\downarrow v_2.$

STS: $(v_1,v_2) \in V[\sigma\longrightarrow\tau]\rho.$

Let $(v'_1,v'_2) \in V[\sigma]\rho$ be given.

TS: $(v_1\ v'_1,\ v_2\ v'_2) \in E[\tau]\rho$

$\Leftarrow$ $(e_1\ v'_1,\ e_2\ v'_2) \in E[\tau]\rho$

$\Leftarrow$(†)$(v'_1,v'_2) \in E[\sigma]\rho$

$\Leftrightarrow$ $(v'_1,v'_2) \in V[\sigma]\rho.$

Q.E.D.

Lemma (Type applications):

$(e_1,e_2) \in E[\forall\alpha.\tau]$ iff $e_1\downarrow \wedge e_2\downarrow \wedge$

$\forall\sigma_1,\sigma_2 \in \mathrm{CTyp}.\ \forall R \in \mathrm{Cand}.\ (e_1\ \sigma_1,\ e_2\ \sigma_2) \in E[\tau](\rho,\alpha\mapsto R).$

Proof: Omitted.

Notation:

Rather than write $E[\cdots\alpha\cdots]\alpha{\mapsto}R$,
we now write simply $[\cdots R\cdots]$.

Notes:

• These lemmas (especially when applied implicitly) /do/ let you avoid routine (and distracting) jumps between $E[\tau]$ and $V[\tau]$ during proofs. They generate routine (and distracting) termination side-conditions.

• The termination side-conditions matter but seem to fall out naturally during proofs.

• If we want to formalize our new notation, we might bake (identifers for) candidates into the syntax for types.

— Canonical forms theorems

We can use the logical relation to show that closed terms of certain types are contextually equivalent to their canonical forms.

We'll write $e \equiv e'$ for contextual equivalence.

Theorem (Canonical forms for pairs):

If $\vdash e : \tau_1 \times \tau_2$,
then $e \equiv \langle v_1,v_2\rangle : \tau_1 \times \tau_2$
for some v_1, $v_2 : \tau_1, \tau_2$.

The idea is to prove this in two steps.

1 (e's η-expansion at type $\tau_1{\times}\tau_2$ evaluates to a canonical form).

$\vdash e\ [\tau_1{\times}\tau_2]\ \mathrm{pair} \downarrow \langle v_1,v_2\rangle$
for some v_1, v_2

Recall that we proved this using the unary LR; see ./20121013.

2. (e is contextually equivalent to its η-expansion).

$\vdash e \equiv e\ [\tau_1 x \tau_2]\ \mathrm{pair} : \tau_1 \times \tau_2$
$\Leftarrow$ (FTLR)
$\vdash e \approx e\ [\tau_1 x \tau_2]\ \mathrm{pair} : \tau_1 \times \tau_2$.

Notes:

- We're working with closed terms for simplicity. The more general form for open terms is perfectly fine.

- We can do such things for a variety of types. The plan is to start with this proof today, do one for homework, and see what we can do in the future.

— Canonical forms for pairs

Relying implicitly on the preceding "convenience" lemmas, we'll not spell out E vs V. Using the preceding notation, we'll inline the relations.

Lemma:

If $\vdash e : \tau_1 \times \tau_2$,
then $(e, e\ [\tau_1{\times}\tau_2]\ \mathrm{pair}) \in [\tau_1{\times}\tau_2]$.

Recall that $\tau_1{\times}\tau_2 = \forall\alpha.(\tau_1{\longrightarrow}\tau_2{\longrightarrow}\alpha){\longrightarrow}\alpha$.

Proof:

Since e and $e\ [\tau_1{\times}\tau_2]$ pair are both well-typed, they terminate. We'll ignore the termination side-conditions until the end of the proof.

Let $\sigma_1,\sigma_2 \in \mathrm{CTyp}$, $R{\in}\mathrm{Cand}$, $(k_1,k_2){\in}[\tau_1{\longrightarrow}\tau_2{\longrightarrow}R]$.
TS: $(e[\sigma_1]k_1, (e[\tau_1{\times}\tau_2]\mathrm{pair})[\sigma_2]k_2) \in [R]$.

WK: $(e,e) \in [\tau_1{\times}\tau_2]$.
(Aside: WK = "we know".)

Plan (this plan will recur):

1. Pick $S = \{ (v_1,v_2) \mid \cdots \}$ to instantiate α.
2. Show $(k_1,\mathrm{pair}) \in [\tau_1{\longrightarrow}\tau_2{\longrightarrow}S]$.
3. By parametricity, obtain
(*) $(e[\sigma_1]k_1, e[\tau_1{\times}\tau_2]\mathrm{pair}) \in [S]$.
4. Hence,
$(e[\sigma_1]k_1,e[\tau_1{\times}\tau_2]\mathrm{pair}[\sigma_2]k_2) \in [R]$.

Our goal with this proof plan, then, is to choose S such that
(*) $\Rightarrow$ (4).

Pick $S := \{ (v_1,v_2) \mid (v_1,v_2[\sigma_2]k_2]) \in [R] \}$.
Aside from Dave: Observe that
$(e_1,e_2) \in [S] \Longleftrightarrow (e_1,e_2[\sigma_2]k_2) \in [R]$.

STS: $(k_1,\text{pair}) \in [\tau_1 \longrightarrow \tau_2 \longrightarrow S]$.

Let $(v'_1,v'_2) \in [\tau_1]$ and $(v''_1,v''_2) \in [\tau_2]$
TS: $(k_1\ v'_1\ v''_1,\ \text{pair}\ v'_2\ v''_2) \in [S]$
$\Leftarrow$ $(k_1\ v'_1\ v''_1,\ \text{pair}\ v'_2\ v''_2\ [\sigma_2]\ k_2) \in [R]$.

WK: pair $v'_2\ v''_2\ [\sigma_2]\ k_2 \mapsto_* k_2\ v'_2\ v''_2$.
By assumption,
$(k_1\ v'_1\ v''_1,\ k_2 \mapsto_* k_2\ v'_2\ v''_2) \in [R]$.

We're not quite done. We implicitly used the convenient lemmas with termination side-conditions. To show $(k_1,\text{pair}) \in [\tau_1 \longrightarrow \tau_2 \longrightarrow S]$, we also have to show the intermediate steps terminate.

For example, to show $(k_1\ v'_1,\ \text{pair}\ v'_2) \in [\tau_1 \longrightarrow S]$, we have to show $k_1\ v'_1$ and pair v'_2 terminate.

WK: $(k_1,k_2) \in [\tau_1 \longrightarrow \tau_2 \longrightarrow R]$
$\Rightarrow$ $(k_1\ v'_1,\ k_2\ v'_2) \in [\tau_2 \longrightarrow R]$.
$\Rightarrow$ $k_1\ v'_1\downarrow$.

WK: pair $v'_2\downarrow$ (by the definition).

Q.E.D.

Why the subtle termination conditions?

Here's one reason: Our model is untyped. (Aside: Such models are sometimes called “realizability models”.) Had we built our model from syntactically well-typed terms (à la Pitts), we would automatically get “M in the LR” ⇒ “M syntactically well-typed” ⇒ “M terminates”.

Why did we build an untyped model? The advantage of the untyped model: We never have to worry about syntactic typing side-conditions. (In Derek's experience, they create a lot of uninteresting proof obligations.)

— Canonical forms theorems for other types

Derek has proven similar results for existential and sum types.

Some relevant papers:

• Logic for parametric polymorhpism. Plotkin and Abadi, TLCA 1993. They work in a logic for reasoning about the model rather than in the model. They go through the statements of these theorems, but not the proofs.

• Categorical models for Abadi and Plotkin's logic for parametricity. Birkedal and Møgelberg, MSCS 2005. They give detailed proofs in the logic for the Plotkin/Abadi theorems.

— Canonical forms for natural numbers

Let's try the same thing for natural numbers.

Lemma:
If $\vdash$ e : nat,
then (e, e [nat] zero succ) ∈ [nat]
where
nat := $\forall\alpha.\alpha \longrightarrow (\alpha\longrightarrow\alpha) \longrightarrow \alpha$
zero : nat = $\Lambda\alpha.\lambda z.\lambda s.z$
succ : nat ⟶ nat = $\lambda n.\Lambda\alpha.\lambda z.\lambda s.s(n\ \alpha\ z\ s)$.

Proof:
Let $\sigma_1,\sigma_2 \in$ CTyp, R∈Cand, $(z_1,z_2)\in[R]$, $(s_1,s_2)\in[R\longrightarrow R]$.
TS: (e σ_1 z_1 s_1, e nat zero zucc σ_2 z_2 s_2) ∈ [R].

WK: (e,e) ∈ [nat].
Plan:
1. Pick S := { (v_1,v_2) | $(v_1,v_2\ \sigma_2\ z_2\ s_2) \in [R]$ } to instantiate α.
1½. Aside from Dave: Observe
$(e_1,e_2) \in [S] \Longleftrightarrow (e_1,e_2\ \sigma_2\ z_2\ s_2) \in [R]$.
2. Show $(z_1,\text{zero}) \in S \wedge (s_1,\text{succ}) \in [S \longrightarrow S]$.
3. Obtain (e σ_1 z_1 s_1, e nat zero zucc) ∈ [S].
4. Hence (e σ_1 z_1 s_1, e nat zero zucc σ_2 z_2 s_2) ∈ [R].

STS: $(z_1,\text{zero}) \in [S] \wedge (s_1,\text{succ}) \in [S \longrightarrow S]$.

(First conjunct):
TS: $(z_1,\text{zero}\ \sigma_2\ z_2\ s_2) \in [R]$
β-reduce zero. Conclude $(z_1,\text{zero}\ \sigma_2\ z_2\ s_2) \in [R]$.

(Second conjunct):
Let $(v_1,v_2) \in [S]$.
TS: $(s_1\ v_1, \text{succ}\ v_2) \in [S]$.

STS: $(s_1\ v_1, \text{succ}\ v_2\ \sigma_2\ z_2\ s_2) \in [R]$.
β-reduce succ. Obtain $(s_1\ v_1, s_2(v_2[\sigma_2]\ z_2\ s_2)) \in [R]$.
STS: $(v_1,v_2\ \sigma_2\ z_2\ s_2) \in [R] \Leftarrow (v_1,v_2) \in [S]$.

The proof is great modulo possible bugs wrt
termination conditions. (There should be no difficult ones).
Q.E.D.

— Homework for Thursday

Do one of these proofs (eg, for sum types):

$$\vdash e : \tau_1 + \tau_2 \Longrightarrow (e, e\ [\tau_1{+}\tau_2]\ \text{inj}_1\ \text{inj}_2) \in [\tau_1{+}\tau_2].$$

— Aside from Dave

In our proof procedure for these canonical forms lemmas, we picked a candidate

$$S := \{\ (v_1,v_2) \mid P(v_1,v_2)\ \}$$

where the predicate P is a predicate on pairs of expressions (rather than values). We then implicitly used the lemma

$$(e_1,e_2) \in [S] \Longleftrightarrow P(e_1,e_2).$$

Such lemmas should be made explicit in case we later attempt to "reuse" our proof in a more complicated model. It's not immediately obvious they'd continue to hold in, say, a step-indexed model.