# Type Theory Essentials

# 1. Common Notations

## 1.1 (meta)variable Names

These are common names used for (meta)variables. Typically, a whole set of characters is used to refer to specific kinds of things. This is similar to how in math, $m$ and $n$ often refer to natural numbers, $x$, $y$, and $z$ often refer to real numbers, etc. Or how in imperative programming, $i$ and $j$ are used for loop variables, array indexes, etc.

| | |
|---|---|
| $M, N, P$ | Terms/Expressions, Proofs |
| $x, y, z$ | Variables (esp. in terms) |
| $a, b, c$ | Variables (esp. in Types) |
| $A, B, C$ | Types, Propositions |
| $\Gamma, \Delta, \Sigma$ | Contexts, Collections of free variables |
| $J$ | Judgments |
| $U, V$ | Values |
| $K, k$ | CK-machine contexts/call stacks, continuations |
| $\sigma$, $\rho$ (rho) | CK-machine value stacks |
| $E, \eta, e$ | Environments |
| $M$ | CK-machine machine state |

## 1.1 (Meta)variable Names (cont.)

| | |
|---|---|
| $\mathcal{D}, \mathcal{E}, \mathcal{F}$ | Proofs / Proof Trees (often inside $\triangledown$ or next to $\vdots$) |
| $\phi, \psi$ | Propositions |
| $t$ | times |
| $w$ | worlds |

Also be aware that font choice matters! In the above table, the metavariable names $\mathcal{M}$ and $\mathsf{M}$ are both capital m's, but use different fonts, and therefore mean different things!

It's not uncommon to see both in the same setting, such as in grammars like so:

one the other

$$\begin{array}{lrl} \text{Machine} & \mathsf{M} ::= & K \triangleright \mathcal{M} \\ & \mid & K \triangleleft V \end{array}$$

or in schematized proofs:

$$\begin{array}{c} \vdots\,\mathcal{D} \\ A \wedge D \ \text{true} \end{array} \qquad \begin{array}{c} \triangledown\,\mathcal{E} \\ E \wedge F \ \text{false} \end{array}$$



1.1 (Meta)variable Names (cont.)

# 1.2 Judgements

Proof theory and type theory uses judgements (or judgments no e) to talk about things. For instance, "$3=2$" is a judgment about two numbers being equal. This one happens to be un-provable. Judgements are objects of knowledge to the prover or user of a logic.

Here are some common judgements and their English! meanings presented in a standard form:

| | |
|---|---|
| $\boxed{A \text{ true}}$ | The proposition $\underline{A}$ is true. |
| $\boxed{A \text{ false}}$ | The proposition $\underline{A}$ is false. |
| $\boxed{A \text{ prop}}$ | The syntactic form $\underline{A}$ is a well-formed proposition. |
| $\boxed{\mathcal{M} \text{ term}}$ | The syntactic form $\underline{\mathcal{M}}$ is a well-formed term. |
| $\boxed{\mathcal{V} \text{ val}}$ | The syntactic form $\underline{\mathcal{V}}$ is a well-formed value. |
| $\boxed{\mathcal{M} : A}$ | The term $\underline{\mathcal{M}}$ proves prop $\underline{A}$ to be true. Sometimes also more generally/precisely: $\mathcal{M} : A \text{ true}$ |
| $\boxed{\mathcal{M} : J}$ | The term $\underline{\mathcal{M}}$ proves judgement $\underline{J}$. |
| $\boxed{\mathcal{M} \Downarrow \mathcal{V}}$ | Term $\underline{\mathcal{M}}$ normalizes to value $\underline{\mathcal{V}}$ |

Judgements can be higher-order, and contain other judgements, as in the case of $M : A$ true,

Another common form of judgement is called a <u>hypothetical</u> judgement, which is a judgment with a collection of hypothesized judgments, usually written

$\boxed{\Gamma \vdash J}$ Under hypotheses $\Gamma$, judgment $J$ is provable.

In such settings, $\Gamma$ itself contains judgment of the same sort as $J$, and $J$ is usually <u>not</u> a hypothetical, but rather something more like $A$ true or $M : A$.

<u>EXERCISE</u>

Identify the judgments in the following:

$$x : A, y : A, x = y \vdash (y \cong x) \text{ true}$$

There are <u>5</u>!

## 1.3 Inference Rule Definitions

Inference rules are defined by writing 0 or more judgements in a row, a line below them which typically has a name written next to it (often right side), and then below that another judgement:

$$\frac{J_0 \qquad J_1 \qquad \cdots \qquad J_n}{J_c}\ \text{rule name}$$

The judgement above the line are called <u>Premises</u> of the rule and the judgment below is called the <u>Conclusion</u> of the rule.

$$\frac{\overbrace{J_0 \qquad J_1 \qquad \cdots \qquad J_n}^{\text{premises}}}{\underbrace{J_c}_{\text{conclusion}}}\ \text{rule name}$$

Judgements in rules <u>almost</u> always have metavariables in them, for example $A$ and $B$ in

$$\frac{A \text{ true} \qquad B \text{ true}}{A \wedge B \text{ true}}\ \wedge I$$

These metavariables indicate that the inference rule can be used for <u>anything</u> that the metavariable could stand for. In the above rule $\underline{\wedge I}$, the metavars $A$ and $B$ can be "instantiated" to be any proposition we like, when we actually use the rule in a proof.

These metavars are constrained tho: if the same metavar shows up multiple times, it <u>must</u> be used for / instantiated at the <u>same</u> thing!

For example, this is a correct use of the rule $\wedge I$:

$$\frac{\text{Foo true} \qquad \text{Bar true}}{\text{Foo} \wedge \text{Bar true}} \wedge I$$

This is <u>not</u>:

A := Foo here

$$\frac{\text{Foo true} \qquad \text{Bar true}}{\text{Quux} \wedge \text{Bar true}} \wedge I$$

but A:= Quux here

<u>BAD</u>!!



1.3 Inference Rule Definitions (cont.)

Sometimes you'll see things that look like inference rules but using dashed lines, for example:

$$\frac{\Gamma \vdash M : A}{\Gamma, x{:}B \vdash M : A}\ wk$$

These technically <u>aren't</u> inference rules, but rather descriptions of properties of the logic. The above example can be read as saying that if $\Gamma \vdash M{:}A$ is provable, then so is $\Gamma, x{:}B \vdash M : A$.

Sometimes you'll also see things that look like inference rules using double lines, for instance

$$\frac{A \wedge B \text{ true}}{B \wedge A \text{ true}}$$

This means both of these:

$$\frac{A \wedge B \text{ true}}{B \wedge A \text{ true}} \qquad \frac{B \wedge A \text{ true}}{A \wedge B \text{ true}}$$



1.3 Inference Rule Definitions (cont.)

## 1.4 Proofs

A proof is a 2-dimensional tree structure made from judgments and inference rule names.

They look like trees where the branches are composed of inference rules instantiated at specific choices of metavars. For example:

$$\dfrac{\dfrac{Foo \wedge Bar \text{ true}}{Bar \text{ true}}\,{\wedge}E_2 \qquad \dfrac{Foo \wedge Bar \text{ true}}{Foo \text{ true}}\,{\wedge}E_1}{Bar \wedge Foo \text{ true}}\,{\wedge}I$$

Each local sub-tree/branch is labeled below by a judgment, and to the side by a rule name. The local subtrees are

$$\dfrac{Foo \wedge Bar \text{ true}}{Bar \text{ true}}\,{\wedge}E_2 \qquad \dfrac{Foo \wedge Bar \text{ true}}{Foo \text{ true}}\,{\wedge}E_1 \qquad \dfrac{Bar \text{ true} \quad Foo \text{ true}}{Bar \wedge Foo \text{ true}}\,{\wedge}I$$

A local subtree is a <u>use</u> of an inference rule. It's a <u>correct</u> use if and only if the local subtree can be formed by instantiating the specified rule's metavariables somehow.

For example, in the above proof, the last local subtree is formed by instantiating the rule $\wedge I$ by choosing

$$A := \text{Bar} \qquad B := \text{Foo}$$

A proof is <u>valid</u> if all local subtrees are formed from correct uses of some inference rule or other in the logic that the proof is using.

Some proofs contain judgements at the leaves, i.e. the judgements are <u>not</u> the conclusions of any inference rule use. These are sometimes called hypotheses or assumptions. Sometimes they are called unproven judgments or proof obligations or goals/subgoals. They represent incomplete parts of a proof.

Any proof with <u>no</u> unproven judgments is a complete proof and it proves its root/bottom-most jugement. We call such proofs <u>sound</u>.

EXERCISE

Consider these rules:

$\boxed{n \text{ nat}}$ $n$ is a natural number.

$$\frac{}{0 \text{ nat}}\,\text{zero}$$

$$\frac{n \text{ nat}}{n' \text{ nat}}\,\text{successor}$$

$$\frac{m \text{ nat} \quad n \text{ nat}}{m+n \text{ nat}}\,\text{plus}$$

1. Prove that $0'' \text{ nat}$ holds.
2. Does $1 \text{ nat}$ hold? Why or why not?
3. How many premises does each rule have? How many conclusions?
4. Is this a valid proof? Why or why not? Is it sound? Why or why not?

$$\cfrac{\cfrac{n \text{ nat}}{n' \text{ nat}}\,\text{successor}}{n'' \text{ nat}}\,\text{successor}$$

5. How about this? Why or why not?

$$\cfrac{\cfrac{}{0 \text{ nat}}}{\cfrac{0 \text{ nat}}{0''' \text{ nat}}\,\text{successor-2}}\,\text{successor}$$



1.4 Proofs (cont.)

# 2. Type Theory Essentials

## 2.1 Judgmental Fragments

Type theories all have a collection of judgements, and a collection of inference rule-like meta properties that define the meanings of the judgements formally. This is called the judgmental fragment.

For example, here is the judgmental fragment for intuitionistic propositional logic:

$\boxed{\Gamma \vdash A \text{ true}}$ The proposition $A$ is true, assuming all of $\Gamma$ is true.

$$\frac{}{\Gamma, A \text{ true} \vdash A \text{ true}} \text{ reflexivity (aka refl)}$$

$$\frac{\Gamma \vdash A \text{ true}}{\Gamma, B \text{ true} \vdash A \text{ true}} \text{ weakening (aka wk)}$$

$$\frac{\Gamma, B \text{ true}, B \text{ true} \vdash A \text{ true}}{\Gamma, B \text{ true} \vdash A \text{ true}} \text{ contraction (aka cn)}$$

$$\frac{\Gamma, A \text{ true}, B \text{ true}, \Gamma' \vdash C \text{ true}}{\Gamma, B \text{ true}, A \text{ true}, \Gamma' \vdash C \text{ true}} \text{ exchange (aka ex)}$$

$$\frac{\Gamma \vdash A \text{ true} \qquad \Gamma, A \text{ true} \vdash B \text{ true}}{\Gamma \vdash B \text{ true}} \text{ transitivity (aka trans)}$$

# 2.1 Judgmental Fragments (cont.)

One absolutely important aspect of a judgmental fragment is that only judgments are distinguished in the meta-rules/properties.

For instance, above we see reference to two kinds of judgment, $\Gamma \vdash J$ and $A$ true. We also see reference to propositions $A$, $B$, and $C$.

BUT we only distinguish judgments from one another, making explicit distinctions by writing $\Gamma \vdash J$ and $A$ true. We never distinguish between propositions, except via meta variables. So we don't talk about what different propositions exist – we never mention forms like $A \wedge B$ or $A \supset B$ or $\neg A$ or anything that could distinguish one proposition from another.

This is because the judgmental fragment explains the meaning of judgments completely independently from what propositions exist.

## 2.2 Typing Fragments

In proof theory, we have propositional fragments, which are rules that explain the meanings of propositions. Because we're doing type theory, we instead have typing fragments, which explain the meanings of types, and when a term has/inhabits some type.

A rule in the typing fragment is a <u>real</u> rule, not a meta-rule or property, so it has a solid line. There are typically two basic judgements in a typing fragment which intentionally look similar:

$\boxed{M : A}$ Term $\underline{M}$ has type $\underline{A}$.

$\boxed{x : A}$ Variable $\underline{x}$ has type $\underline{A}$.

Usually, the first is used as the "succedent" of a hypothetical judgment, while the second is what fills the context, which is called the "antecedent" or "hypotheses", etc.

$$\boxed{\Gamma \vdash M : A}$$

The succedent of the hypothetical judgment

looks like $x : B, y : C, z : D, \ldots$
aka context, antecedent, hypotheses, assumptions

## 2.2 Typing Fragments (cont.)

Here is an example typing fragment for a very simple type theory:

$\boxed{\Gamma \vdash M : A}$ Term $\underline{M}$ has type $\underline{A}$ in free variable context $\underline{\Gamma}$.

$$\frac{\Gamma \vdash M : A \qquad \Gamma \vdash N : B}{\Gamma \vdash \langle M, N \rangle : A \times B}\times I \qquad \frac{\Gamma \vdash P : A \times B}{\Gamma \vdash \mathsf{fst}\, P : A}\times E_1$$

$$\frac{\Gamma \vdash P : A \times B}{\Gamma \vdash \mathsf{snd}\, P : B}\times E_2$$

Rules in a typing fragment <u>often</u> but not always involve type formers such as "$\times$" here. They almost always involve term formers like "$\langle -,- \rangle$", "fst_", and "snd_".

Here is a typing rule with no type formers:

$$\frac{\Gamma \vdash M : A \qquad \Gamma, x : A \vdash N : B}{\Gamma \vdash \mathsf{let}\ x = M\ \mathsf{in}\ N \ :\ B}\mathsf{let}$$

EXERCISE

Consider the following rules:

$\boxed{\Delta \vdash A \text{ nec}}$ — $A$ is neccessary under assumptions $\Delta$

$\boxed{\Delta ; \Gamma \vdash A \text{ true}}$ — $A$ is true under assumptions $\Delta$ and $\Gamma$

$$\frac{\Delta \vdash A \text{ nec}}{\Delta ; \Gamma \vdash A \text{ true}}\ \text{nec-true}$$

$$\frac{\Delta ; \emptyset \vdash A \text{ true}}{\Delta \vdash A \text{ nec}}\ \text{true-nec}$$

$$\frac{}{\Delta ; \Gamma \vdash A \text{ nec}}\ \text{nechyp } (A \text{ nec} \in \Delta)$$

$$\frac{}{\Delta ; \Gamma \vdash A \text{ true}}\ \text{truehyp } (A \text{ true} \in \Gamma)$$

$$\frac{\Delta ; \Gamma \vdash \Box A \text{ true} \qquad \Delta ; \Gamma, A \text{ true} \vdash B \text{ true}}{\Delta ; \Gamma \vdash B \text{ true}}\ \Box 1$$

$$\frac{\Delta ; \Gamma \vdash A \text{ nec}}{\Delta ; \Gamma \vdash \Box A \text{ true}}\ \Box 2$$

EXERCISE (cont.)

1. Which rules are part of the judgmental fragment? Why?

2. Which rules are part of the typing/propositional fragment? Why?

## 2.3 When is a typing fragment "ok"?

A fundamental part of modern proof theory and type theory is the idea that we ought to justify writing a propositional or typing fragment in some way. Failure to do so might, and often does, lead to logics that are at best useless, and at worst, deceptive and inconsistent.

The most common way of doing this is by using two kinds of meta properties of a typing fragment that we call local soundness and local completeness. If we can show that a typing fragment's rules are both locally sound and complete, then we have shown that they don't introduce any bad inferences that contradict what the judgemental fragment says.

### 2.3.1 Local Soundness aka $\beta$-reduction

Suppose that we're using the following two-rule typing fragment:

$$\frac{M : A \qquad N : B}{\langle M, N \rangle : A \times B} \times I \qquad\qquad \frac{P : A \times B}{\mathrm{fst}\ P : A} \times E$$

## 2.3.1 Local Soundness aka β-reduction (cont.)

To use local soundness as a check on this typing fragment, we must first be able to divide the rules into groups for each connective or type former. Such groups must only involve the one specified type former, or else have no type formers at all.

Then for each group of rules for a given type former, we must divide the the rules into "introduction" rules (intros), and "elimination" rules (elims). Intros must have exactly one occurrence of the type former in the conclusion, and elims must have atleast one in the premises.

Our rule set has only rules for ×, so the grouping is trivial. It can also be divided into intros and elims. The rule names have been chosen to reflect this division: I for intro rules, E for elims. This is often done, but not required.

Group for ×

$$\underbrace{\frac{M:A \qquad N:B}{\langle M,N\rangle : A\times B}\times I}_{\text{intros}} \qquad\qquad \underbrace{\frac{P : A\times B}{\text{fst } P : A}\times E}_{\text{elims}}$$



2.3.1 Local Soundness aka β-reduction (cont.)

## 2.3.1 Local Soundness aka $\beta$-reduction (cont.)

The next step to using local soundness checks is to show that for any proof that ends with an intro ruled followed immediately by an elim, we can reconstruct a proof from the rest of the proof, which does not use intros or elims, but which still leads to the same type context, etc. except term

The proof terms do not have to be the same at the end, however. We may use any judgmental rules we want to do this.

For our example rules, there's only one intro and one elim, so we must show the above transformation for only the following shape of proofs:

$$\dfrac{\dfrac{\begin{matrix}\vdots\,\mathcal{D} \\ M : A\end{matrix} \qquad \begin{matrix}\vdots\,\mathcal{E} \\ N : B\end{matrix}}{\langle M, N \rangle : A \times B}\times I}{\mathrm{fst}\langle M, N\rangle : A}\times E$$

We must construct a (new?) proof of $P : A$, for some term $P$, using only $\begin{matrix}\vdots\,\mathcal{D} \\ M : A\end{matrix}$ and $\begin{matrix}\vdots\,\mathcal{E} \\ N : B\end{matrix}$, and the judgmental rules.

## 2.3.1 Local Soundness aka $\beta$-reduction (cont.)

Can we do this? Yes! We actually <u>already</u> have a proof of $P : A$ for some $P$, namely the proof

$$\begin{array}{c} \vdots\, \mathcal{D} \\ M : A \end{array}$$

So we "witness" the local soundness of these rules by writing

$$\dfrac{\dfrac{\begin{array}{c} \vdots\, \mathcal{D} \\ M : A \end{array} \qquad \begin{array}{c} \vdots\, \mathcal{E} \\ N : B \end{array}}{\langle M, N \rangle : A \times B}\times I}{\mathsf{fst}\,\langle M, N \rangle : A}\times E \quad \Rightarrow \quad \begin{array}{c} \vdots\, \mathcal{D} \\ M : A \end{array}$$

Because the structure of terms is often similar to the structure of proofs, we can typically reconstruct the proof given just the term. In such cases, we often just write the above witness like so:

$$\mathsf{fst}\,\langle M, N \rangle \Rightarrow_\beta M$$

In this form, we usually call such a thing a "$\beta$-reduction".

For rules without type formers, we still must witness local soundness, but this time for each rule in isolation.

Consider the rule for "let" that was mentioned a while back:

$$\frac{\Gamma \vdash M : A \qquad \Gamma, x : A \vdash N : B}{\Gamma \vdash \text{let } x = M \text{ in } N : B} \text{let}$$

How might we justify this with local soundness? Well, we would need to find some way to reconstruct a proof of $\Gamma \vdash P : B$ for some $P$, without using "let" itself.

Suppose that our judgmental fragment had the following rule which looks <u>very</u> similar to "let" but which is formally distinct:

$$\frac{\Gamma \vdash M : A \qquad \Gamma, x : A \vdash N : B}{\Gamma \vdash [M/x]N : B}$$

Then the $\beta$-reduction is easy! It's just:

$$\text{let } x = M \text{ in } N \Rightarrow_\beta [M/x]N$$



2.3.1 Local soundness aka $\beta$-reduction (cont.)

EXERCISE

Is the following set of typing rules locally sound? Why or why not?

$$\frac{M : A}{\text{left } M : A \text{ tonk } B}\ \text{tonk-I} \qquad \frac{N : A \text{ tonk } B}{\text{snd } N : B}\ \text{tonk-E}$$

EXERCISE

Is the following set of typing rules locally sound? Why or why not?

$$\frac{M : A \qquad N : B}{\langle M, N \rangle : A \text{ plonk } B} \qquad \frac{P : A \text{ plonk } B}{\text{snd } P : B}$$

EXERCISE

Prove that this set of typing rules is locally sound. What judgmental rules do you need in order to do this?

$$\frac{\Gamma \vdash M : A \qquad \Gamma \vdash N : B}{\Gamma \vdash \langle M, N \rangle : A \otimes B}\ \otimes I$$

$$\frac{\Gamma \vdash P : A \otimes B \qquad \Gamma, x : A, y : B \vdash N : C}{\Gamma \vdash \text{split } P \text{ as } \langle x, y \rangle \text{ in } N : C}\ \otimes E$$



2.3.1 Local Soundness aka β-reduction (cont.)

2.3.2. Local Completeness aka $\eta$-expansion

The second component of modern checks on a typing fragment's ok-ness is local completeness aka $\eta$-expansion. Like $\beta$-reduction, we must divide rules by type former, and for each group of rules, we must divid into intros and elims.

eta

To witness local completeness, we sort of do the reverse of local soundness: We show that for any arbitrary proof of the connective in question, we can use each elim on it, followed by using each intro, and reconstruct a term with the same initial type. Again we can also use any judgmental rules we like.

Let's try this for the two rules $\times I$ and $\times E$. Given an arbitrary proof like so:

$$\begin{array}{c} \vdots \mathcal{D} \\ P : A \times B \end{array}$$

We must elim then intro to get a new proof of $P' : A \times B$ for some $P'$.

Well, there's one elim: $\times E$, so let's use it:

$$\dfrac{\begin{array}{c} \vdots \mathcal{D} \\ P : A \times B \end{array}}{\mathrm{fst}\, P : A} \times E$$

This is good so far. But there's only one intro — $\times I$ — and it has two premises:

$$\dfrac{M : A \qquad N : B}{\langle M, N \rangle : A \times B} \times I$$

Since we're trying to make a proof of $P' : A \times B$, we need to have a proof of $M : A$ and a proof of $N : B$. We <u>do</u> have a proof of $M : A$, namely

$$\dfrac{\begin{array}{c} \vdots \mathcal{D} \\ P : A \times B \end{array}}{\mathrm{fst}\, P : A} \times E$$

But we don't have a proof of $N : B$ for some $N$, <u>nor</u> can we get one!

Since there is no way to make a proof of $N : B$, there is also no way to do $\eta$-expansion and so these rules are <u>NOT</u> locally complete!

## 2.3.2 Local Completeness aka $\eta$-expansion (cont.)

What's missing from our rules is the second elim. Let's add it:

$$\frac{M : A \qquad N : B}{\langle M, N \rangle : A \times B}\,\times I \qquad\qquad \frac{P : A \times B}{\mathsf{fst}\ P : A}\,\times E_1$$

$$\frac{P : A \times B}{\mathsf{snd}\ P : B}\,\times E_2$$

Now we can easilly witness local completeness:

$$\begin{array}{c} \vdots\,\mathcal{D} \\ P : A \times B \end{array} \quad \Longrightarrow \quad \dfrac{\dfrac{\begin{array}{c} \vdots\,\mathcal{D} \\ P : A \times B \end{array}}{\mathsf{fst}\ P : A}\,\times E_1 \qquad \dfrac{\begin{array}{c} \vdots\,\mathcal{D} \\ P : A \times B \end{array}}{\mathsf{snd}\ P : B}\,\times E_2}{\langle \mathsf{fst}\ P, \mathsf{snd}\ P \rangle : A \times B}\,\times I$$

When we can omit the full proofs because of structural similarity, we write

$$P : A \times B \ \Longrightarrow_{\eta}\ \langle \mathsf{fst}\ P, \mathsf{snd}\ P \rangle$$

The type annotation can be omitted when its obvious from the written context, but often its useful to let people know the type former that matters.

EXERCISE

1. Is $\underline{\text{tonk}}$ from 2.3.1 locally complete? Why or why not?
2. Is $\underline{\text{plonk}}$? Why or why not?
3. Is $\underline{\otimes}$? Why or why not?



2.3.2 Local Completeness aka $\eta$-expansion (cont.)

## 2.3.3 Harmony

When a set of typing rules is both locally sound and locally complete, we say its rules are in harmony. A type theory which exhibits harmony is considered "ok" to use. One which does not should be considered suspect: not guaranteed to be bad, but definitely not guaranteed to be good either.

The purpose of harmony is to show that the typing fragment doesn't alter the inferences that are possible, except in ways that fundamentally involve the new type formers that the typing fragment defines. The witness in local soundness and completeness all transform the proofs in ways that are agnostic to the type formers except the one of interest. That means that they show that anything not involving those type formers must have been left unaltered.

## 2.3.4 Judgmental Fragments Again

When we last discussed judgmental fragments, we mentioned that they're not exclusively actual inference rules, but rather also are meta rules, or properties of the whole logic. It's now time to explain what that means in detail.

Let's suppose that our judgmental fragment looks like this:

$$\frac{}{\Gamma, x:A \vdash x:A}\ \text{refl}$$

$$\frac{\Gamma \vdash M:A}{\Gamma, y:B \vdash M:A}\ \text{wk (assuming } y \text{ is not in } \Gamma)$$

$$\frac{\Gamma, x:B, y:B \vdash M:A}{\Gamma, x:B \vdash [x/y]M:A}\ \text{cn}$$

$$\frac{\Gamma, x:A, y:B, \Gamma' \vdash N:C}{\Gamma, y:B, x:A, \Gamma' \vdash N:C}\ \text{ex}$$

$$\frac{\Gamma \vdash M:A \qquad \Gamma, x:A \vdash N:B}{\Gamma \vdash [M/x]N:B}\ \text{subst}$$

What does it meant to say that the typing fragment has these as properties?

It means precisely that for any arbitrary proof using rules in the typing fragment alone, we can show that the properties – treated as rules – can be "push" down into the proof, and eventually be made to vanish entirely.

Let's define a typing fragment for reference:

$$\frac{\Gamma \vdash M : A \qquad \Gamma \vdash N : B}{\Gamma \vdash \langle M, N \rangle : A \times B} \times I$$

$$\frac{\Gamma \vdash P : A \times B}{\Gamma \vdash \mathrm{fst}\ P : A} \times E_1$$

$$\frac{\Gamma \vdash P : A \times B}{\Gamma \vdash \mathrm{snd}\ P : B} \times E_2$$

$$\frac{\Gamma \vdash M : A \qquad \Gamma, x : A \vdash N : B}{\Gamma \vdash \mathrm{let}\ x = M\ \mathrm{in}\ N : B} \mathit{let}$$

$$\frac{}{\Gamma \vdash x : A} \mathit{var}\ (\text{assuming } x : A \in \Gamma)$$

## 2.3.4 Judgmental Fragments Again (cont.)

Let's now show that these 5 rules exhibit the property "wk" (aka weakening).

Suppose we have an arbitrary proof that ends with the rule $\times I$; then the fake rule wk:

$$\dfrac{\dfrac{\begin{array}{c}\vdots\,\mathcal{D}\\ \Gamma\vdash M:A\end{array}\qquad\begin{array}{c}\vdots\,\mathcal{E}\\ \Gamma\vdash N:B\end{array}}{\Gamma\vdash\langle M,N\rangle:A\times B}\,\times I}{\Gamma,y:C\vdash\langle M,N\rangle:A\times B}\,\text{wk (}y\text{ not in }\Gamma)$$

We can push the use of wk up like so:

$$\dfrac{\dfrac{\begin{array}{c}\vdots\,\mathcal{D}\\ \Gamma\vdash M:A\end{array}}{\Gamma,y:C\vdash M:A}\,\text{wk}(y\text{ not in }\Gamma)\qquad\dfrac{\begin{array}{c}\vdots\,\mathcal{E}\\ \Gamma\vdash N:B\end{array}}{\Gamma,y:C\vdash N:B}\,\text{wk }(y\text{ not in }\Gamma)}{\Gamma,y:C\vdash\langle M,N\rangle:A\times B}\,\times I$$

Sometimes instead of $\dfrac{\begin{array}{c}\vdots\,\mathcal{D}\\ \Gamma\vdash M:A\end{array}}{\Gamma,y:C\vdash M:A}\,\text{wk}$ you'll see $\begin{array}{c}\vdots\,\text{wk}(\mathcal{D})\\ \Gamma,y:C\vdash M:A\end{array}$

This means the same thing.



2.3.4 Judgmental Fragments Again (cont.)

## 2.3.4 Judgmental Fragments Again (cont.)

We can similarly push wk up for $\times E_1$ and $\times E_2$:

$$\cfrac{\cfrac{\begin{array}{c}\vdots\,\mathcal{D}\\ \Gamma \vdash p : A \times B\end{array}}{\Gamma \vdash \mathsf{fst}\ p : A}\ \times E_1}{\Gamma, y:C \vdash \mathsf{fst}\ p : A}\ \mathsf{wk}\ (y \text{ not in } \Gamma) \quad \Rightarrow \quad \cfrac{\cfrac{\begin{array}{c}\vdots\,\mathcal{D}\\ \Gamma \vdash p : A \times B\end{array}}{\Gamma, y:C \vdash p : A \times B}\ \mathsf{wk}}{\Gamma, y:C \vdash \mathsf{fst}\ p : A}\ \times E_1$$

$$\cfrac{\cfrac{\begin{array}{c}\vdots\,\mathcal{D}\\ \Gamma \vdash p : A \times B\end{array}}{\Gamma \vdash \mathsf{snd}\ p : B}\ \times E_2}{\Gamma, y:C \vdash \mathsf{snd}\ p : B}\ \mathsf{wk} \quad \Rightarrow \quad \cfrac{\cfrac{\begin{array}{c}\vdots\,\mathcal{D}\\ \Gamma \vdash p : A \times B\end{array}}{\Gamma, y:C \vdash p : A \times B}\ \mathsf{wk}}{\Gamma, y:C \vdash \mathsf{snd}\ p : B}\ \times E_2$$

The var rule is interesting. It's equally simple, but it has no premises, so wk vanishes!

$$\cfrac{\cfrac{}{\Gamma \vdash x : A}\ \mathsf{var}\ (x:A \in \Gamma)}{\Gamma, y:C \vdash x : A}\ \mathsf{wk}\ (y \text{ not in } \Gamma) \quad \Rightarrow \quad \cfrac{}{\Gamma, y:C \vdash x : A}\ \mathsf{var}\ (x:A \in \Gamma, y:C)$$

These rules ($\times I$, $\times E_1$, $\times E_2$, var) all have one thing in common: none of them involve different $\Gamma$'s. But "let" does have a diffent $\Gamma$ in one of its premises. This will be special.



2.3.4 Judgmental Fragments Again (cont.)

Let's now do the same thing for "let":

$$
\dfrac{
\dfrac{\dfrac{\vdots\,\mathcal{D}}{\Gamma \vdash M : A} \qquad \dfrac{\vdots\,\mathcal{E}}{\Gamma, x{:}A \vdash N : B}}{\Gamma \vdash \text{let } x = M \text{ in } N : B}\,\text{let}
}{\Gamma, y{:}C \vdash \text{let } x = M \text{ in } N : B}\,\text{wk } (y \text{ not in } \Gamma)
$$

$$\Downarrow$$

$$
\dfrac{
\dfrac{\dfrac{\vdots\,\mathcal{D}}{\Gamma \vdash M : A}}{\Gamma, y{:}C \vdash M : A}\,\text{wk } (y \text{ not in } \Gamma)
\qquad
\dfrac{\dfrac{\vdots\,\mathcal{E}}{\Gamma, x{:}A \vdash N : B}}{\Gamma, x{:}A, y{:}C \vdash N : B}\,\text{wk } (y \text{ not in } \Gamma, x{:}A)
}{\Gamma, y{:}C \vdash \text{let } x = M \text{ in } N : B}\,\text{let}
$$

whoops! insert ex here:

$$\dfrac{\Gamma, x{:}A, y{:}C \vdash N : B}{\Gamma, y{:}C, x{:}A \vdash N : B}\,\text{ex}$$

Notice! The new use of weakening on the proof $\mathcal{E}$ has a <u>more constrained</u> side condition! We started with $y$ not in $\Gamma$ but we <u>need</u> $y$ not in $\Gamma, x{:}A$. But what if $x$ and $y$ are actually the same variable name?

There are a handful of different ways of resolving this, but the simplest one is to just pretend it never arises in actual proofs, by guaranteeing up front that we never repeat variable names. This is <u>almost</u> sufficient but can have limits.

We have now shown that each rule in the typing fragment permits wk to be pushed upwards. We also can see that the rules that have no premises, namely just var, get rid of the wk entirely.

If our proofs are valid, using only these 5 rules, then they <u>always</u> end at the leaves with a use of the var rule, and so we can push wk all the way up, and it will <u>always</u> vanish from a proof!

So we say that weakening is exhibited by these rules as a set.

Now we do this for the other judgmental rules. If we can show that all the judgmental rules are exhibited by the typing fragment, then our type theory is "ok" (modulo $\beta$ and $\eta$ being witnessed). If not, our type theory is <u>very</u> broken.

EXERCISE

Show that the typing fragment exhibits the judgmental meta-property "cn".

EXERCISE

Show the same for "refl". This one is peculiar, so explain what's odd about it.

EXERCISE

Show the same for "ex".

EXERCISE

Show the same for "subst". Assume variable names are never re-used.

EXERCISE

Given that the subst property uses this notation $[M/x]N$ in the conclusions but no rule in the typing fragment uses this. Therefore, $[M/x]N$ <u>must</u> be meta-notation for some <u>meta operation</u>!

Define $[M/x]N$ as if its a function. I'll start:

$[M/x]\langle N, N'\rangle = \langle [M/x]N, [M/x]N'\rangle$