Università di Genova | DIMA DIPARTIMENTO DI MATEMATICA

Context, judgement, deduction

j.w.w. Ivan Di Liberti

2nd **ItaCa Workshop**

Greta Coraglia

$$\text{(DTy)}\ \frac{\Gamma \vdash a : A \qquad \Gamma.A \vdash B}{\Gamma \vdash B[a]} \qquad \text{(Cut)}\ \frac{x;\Gamma \vdash \phi \qquad x;\Gamma,\phi \vdash \psi}{x;\Gamma \vdash \psi}$$

$$\text{(DTy)}\ \frac{\Gamma \vdash a : A \qquad \Gamma.A \vdash B}{\Gamma \vdash B[a]} \qquad \text{(Cut)}\ \frac{x;\Gamma \vdash \phi \qquad x;\Gamma,\phi \vdash \psi}{x;\Gamma \vdash \psi}$$

$$\text{(DTy)}\ \frac{\Gamma \vdash a : A\ \textbf{Term} \qquad \Gamma.A \vdash B\ \textbf{Type}}{\Gamma \vdash B[a]\ \textbf{Type}} \qquad \text{(Cut)}\ \frac{x;\Gamma \vdash \phi\ \textbf{Form} \qquad x;\Gamma,\phi \vdash \psi\ \textbf{Form}}{x;\Gamma \vdash \psi\ \textbf{Form}}$$

Why does this happen?
How do rules *really* work, syntactically?
What about constructors/connectives?

Propositions as types

$$(\text{DTy})\ \frac{\Gamma \vdash a : A \qquad \Gamma.A \vdash B}{\Gamma \vdash B[a]} \qquad (\text{Cut})\ \frac{x; \Gamma \vdash \phi \qquad x; \Gamma, \phi \vdash \psi}{x; \Gamma \vdash \psi}$$

Propositions as types: it explains the similarities, it doesn't explain *why* these "shapes" in the syntax nor the difference between judgements involving different objects.

[...] so we have constructions acting on constructions.

- William Howard to Philip Wadler

Propositions as types

$$\text{(DTy)}\ \frac{\Gamma \vdash a : A \qquad \Gamma.A \vdash B}{\Gamma \vdash B[a]} \qquad \text{(Cut)}\ \frac{x;\Gamma \vdash \phi \qquad x;\Gamma,\phi \vdash \psi}{x;\Gamma \vdash \psi}$$

Propositions as types: it explains the similarities, it doesn't explain *why* these "shapes" in the syntax nor the difference between judgements involving different objects.

[...] so we have functors acting on functors.

~~– William Howard to Philip Wadler~~

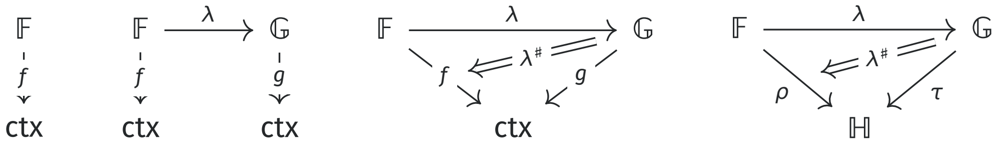
An account of context, judgement, deduction

A *pre-judgemental theory* is specified through the following data:

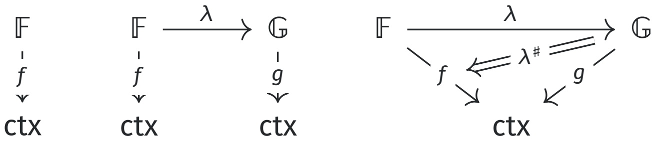
context (ctx) a category (with terminal object $\diamond$);

judgement ($\mathcal{J}$) *judgement classifiers*, a class of functors $f : \mathbb{F} \to \mathrm{ctx}$ over the category of contexts;

deduction ($\mathcal{R}$) *rules*, a class of functors $\lambda : \mathbb{F} \to \mathbb{G}$;
($\mathcal{C}$) *cuts*, a class of 2-dimensional cells filling (some) triangles induced by the rules (functors in $\mathcal{R}$) and the judgements (functors in $\mathcal{J}$).



Categories as syntax



Whenever $F \in f^{-1}(\Gamma)$ we read $\Gamma \vdash F\ \mathbb{F}$.
Whenever $F, F' \in f^{-1}(\Gamma)$ and $F = F'$ we read $\Gamma \vdash F =_{\mathbb{F}} F'$.

$$(\lambda)\ \frac{\Gamma\ \vdash F\ \mathbb{F}}{g\lambda F \vdash \lambda F\ \mathbb{G}}$$

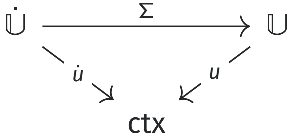
and, possibly, Γ and $g\lambda F$ and related by a map

$$\lambda^{\sharp}_{F} : g\lambda F \to \Gamma$$

Example: toy MLTT

toy MLTT:
$$\begin{cases} \mathsf{ctx} = \text{contexts and substitutions} \\ \mathcal{J} = \{\dot{u}, u\} \\ \mathcal{R} = \{\Sigma\}, \text{ with } \Sigma : (a, A) \longmapsto A \\ \mathcal{C} = \{id : u \circ \Sigma \Rightarrow \dot{u}\} \end{cases}$$

$\dot{u} : \dot{\mathbb{U}} \to \mathsf{ctx}$	$\Gamma \vdash (a, A)\ \dot{\mathbb{U}}$	a is a term of type A in context Γ
$u : \mathbb{U} \to \mathsf{ctx}$	$\Gamma \vdash A\ \mathbb{U}$	A is a type in context Γ



$$(\Sigma)\ \frac{\Gamma \vdash (a, A)\ \dot{\mathbb{U}}}{\Gamma \vdash A\ \mathbb{U}}$$

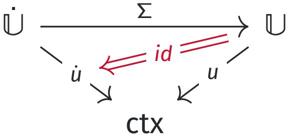
the type of a in context Γ is a type in context Γ

Example: toy MLTT

toy MLTT:

$$\begin{cases} \mathrm{ctx} = \text{contexts and substitutions} \\ \mathcal{J} = \{\dot{u}, u\} \\ \mathcal{R} = \{\Sigma\},\ \text{with } \Sigma : (a, A) \longmapsto A \\ \mathcal{C} = \{id : u \circ \Sigma \Rightarrow \dot{u}\} \end{cases}$$

$\dot{u} : \dot{\mathbb{U}} \longrightarrow \mathrm{ctx}$	$\Gamma \vdash (a, A)\ \dot{\mathbb{U}}$	a is a term of type A in context Γ
$u : \mathbb{U} \longrightarrow \mathrm{ctx}$	$\Gamma \vdash A\ \mathbb{U}$	A is a type in context Γ



$$(\Sigma)\ \frac{\Gamma \vdash (a, A)\ \dot{\mathbb{U}}}{\Gamma \vdash A\ \mathbb{U}}$$

the type of a in context Γ is a type in context Γ

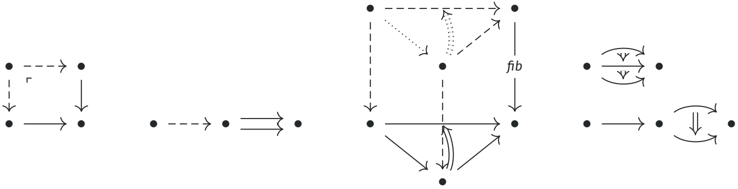
Judgemental theories

This is nice and all, but we can't *do* anything with it.

We express the computational power of a deductive system
into 2-categorical constructions.

A *judgemental theory* $(\mathrm{ctx}, \mathcal{J}, \mathcal{R}, \mathcal{C})$ is a pre-judgemental theory such that

1. $\mathcal{R}$ and $\mathcal{C}$ are closed under composition;
2. the judgements are precisely those rules whose codomain is ctx;
3. $\mathcal{R}$ and $\mathcal{C}$ are closed under *finite limits*, $\sharp$-*liftings*, *whiskering* and *pasting*.



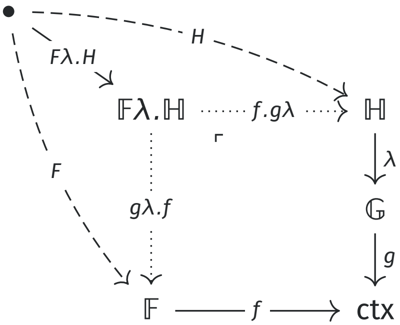
Nested judgements

Pullbacks compute *nested judgements* such as

$$\Gamma \vdash a : A \qquad \Gamma.A \vdash B$$

$$x;\Gamma \vdash \phi \qquad x;\Gamma,\phi \vdash \psi$$

because



$$\Gamma \vdash F\lambda.H\ \mathbb{F}\lambda.\mathbb{H}$$

really is

$$\Gamma \vdash H\ \mathbb{H} \qquad g\lambda H \vdash F\ \mathbb{F}$$

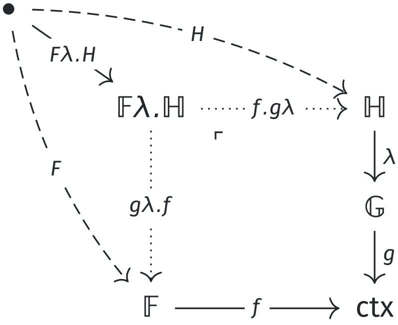
Nested judgements

Pullbacks compute *nested judgements* such as

$$\Gamma \vdash a : A \qquad \Gamma.A \vdash B$$

$$x;\Gamma \vdash \phi \qquad x;\Gamma,\phi \vdash \psi$$

because



$$\Gamma \vdash F\lambda.H\ \mathbb{F}\lambda.\mathbb{H}$$

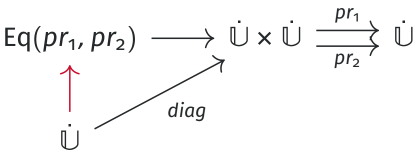
really is

$$\Gamma \vdash H\ \mathbb{H} \qquad g\lambda H \vdash F\ \mathbb{F}$$

Example: toy MLTT

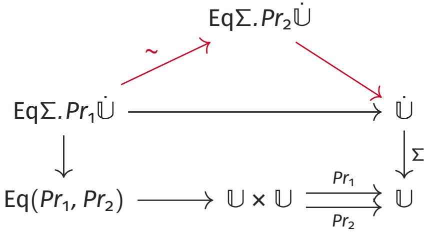
toy MLTT: ctx, $\mathcal{J} = \{\dot{u}, u\}$, $\mathcal{R} = \{\Sigma\}$, $\mathcal{C} = \{id : u \circ \Sigma \Rightarrow \dot{u}\}$

In the judgemental theory generated by (ctx, $\mathcal{J}, \mathcal{R}, \mathcal{C}$) we find the following:



reads as

$$(\rho)\ \frac{\Gamma \vdash (a, A)\ \dot{\mathbb{U}}}{\Gamma \vdash \rho(a, A)\ \mathrm{Eq}(pr_1, pr_2)}$$



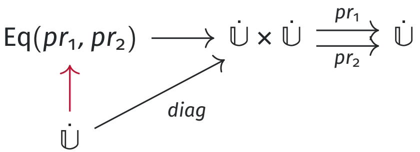
reads as

$$(\sigma)\ \frac{\Gamma \vdash (a, (A, B))\ \mathrm{Eq}\Sigma.Pr_1\dot{\mathbb{U}}}{\Gamma \vdash \sigma(a, (A, B))\ \dot{\mathbb{U}}}$$

Example: toy MLTT

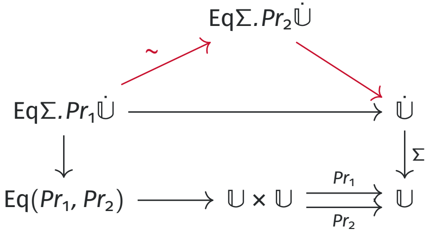
toy MLTT: ctx, $\mathcal{J} = \{\dot{u}, u\}$, $\mathcal{R} = \{\Sigma\}$, $\mathcal{C} = \{id : u \circ \Sigma \Rightarrow \dot{u}\}$

In the judgemental theory generated by (ctx, $\mathcal{J}, \mathcal{R}, \mathcal{C}$) we find the following:



reads as

$$(\rho)\ \frac{\Gamma \vdash a : A}{\Gamma \vdash a =_A a}$$



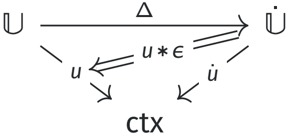
reads as

$$(\sigma)\ \frac{\Gamma \vdash a : A \quad \Gamma \vdash A = B}{\Gamma \vdash a : B}$$

jDTT, I: definition

DTT:

$$\begin{cases} \mathrm{ctx} = \text{contexts and substitutions} \\ \mathcal{J} = \{\dot u, u\}, \text{ with } u, \dot u \text{ fibrations} \\ \mathcal{R} = \{\Sigma, \Delta\}, \text{ with } \Sigma \dashv \Delta \\ \mathcal{C} = \{id : u \circ \Sigma \Rightarrow \dot u, \epsilon, \eta\}, \text{ with } \epsilon, \eta \text{ cartesian} \end{cases}$$



$$\frac{\Gamma \vdash A\ \mathbb{U}}{\dot u \Delta A \vdash \Delta A\ \dot{\mathbb{U}}} \qquad \frac{\Gamma \vdash A}{\Gamma.A \vdash q_A : A\delta_A}$$

jDTT, I: definition

DTT:
$$\begin{cases} \text{ctx} = \text{contexts and substitutions} \\ \mathcal{J} = \{\dot{u}, u\}, \text{ with } u, \dot{u} \text{ fibrations} \\ \mathcal{R} = \{\Sigma, \Delta\}, \text{ with } \Sigma \dashv \Delta \\ \mathcal{C} = \{id : u \circ \Sigma \Rightarrow \dot{u}, \epsilon, \eta\}, \text{ with } \epsilon, \eta \text{ cartesian} \end{cases}$$

Theorem (1)

If $\dot{u}, u$ are discrete fibrations, the judgmental theory generated by DTT is equivalent to a natural model* à la Awodey.

Theorem (2)

The judgmental theory generated by DTT contains codes for all rules of dependent type theory.

*hence categories with families, attributes, etc

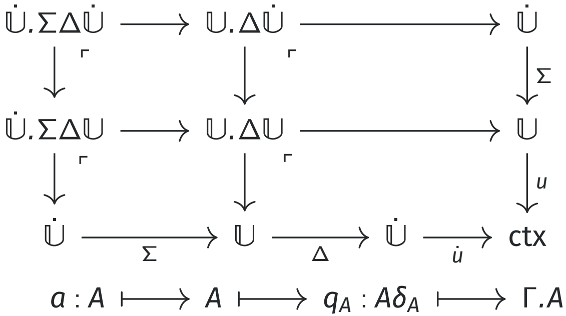
jDTT, II: coding dependent families

$$\Gamma \vdash a : A \qquad \Gamma.A \vdash b : B$$

$$\Gamma \vdash A \qquad \Gamma.A \vdash b : B$$

$$\Gamma \vdash a : A \qquad \Gamma.A \vdash B$$

$$\Gamma \vdash A \qquad \Gamma.A \vdash B$$



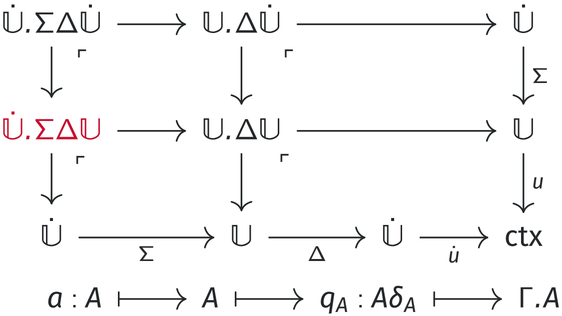
jDTT, II: coding dependent families

$\Gamma \vdash a : A \quad \Gamma.A \vdash b : B$

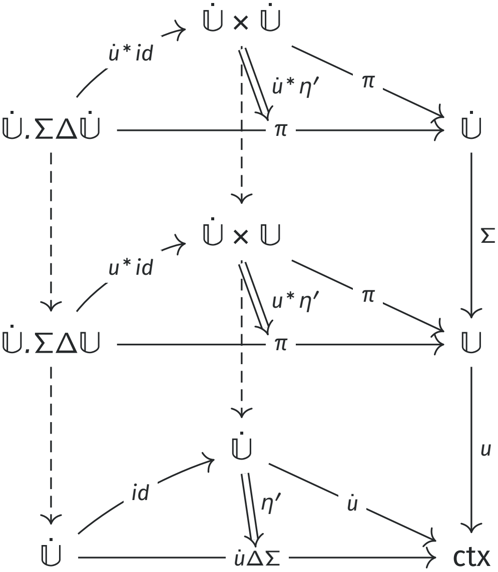
$\Gamma \vdash A \quad \Gamma.A \vdash b : B$

$\Gamma \vdash a : A \quad \Gamma.A \vdash B$

$\Gamma \vdash A \quad \Gamma.A \vdash B$



jDTT, III: type dependency as cuts



$$\frac{\Gamma.A \vdash (a, b)\ \dot{\mathbb{U}}.\Sigma\Delta\dot{\mathbb{U}}}{\Gamma \vdash ??\ \dot{\mathbb{U}}}$$

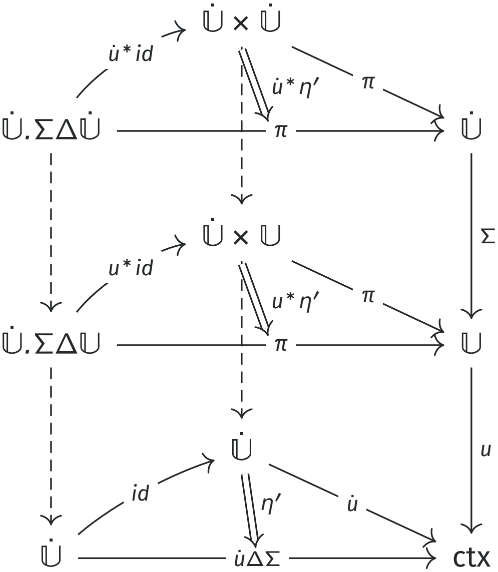
$$?? \longrightarrow b$$

$$\frac{\Gamma.A \vdash (a, B)\ \dot{\mathbb{U}}.\Sigma\Delta\mathbb{U}}{\Gamma \vdash ??\ \mathbb{U}}$$

$$?? \longrightarrow B$$

$$\Gamma \longrightarrow \Gamma.A$$

jDTT, III: type dependency as cuts



$$(\mathrm{DTm})\ \frac{\Gamma \vdash a : A \qquad \Gamma.A \vdash b : B}{\Gamma \vdash b[a] : B[a]}$$

$$b[a] \to b$$

$$(\mathrm{DTy})\ \frac{\Gamma \vdash a : A \qquad \Gamma.A \vdash B}{\Gamma \vdash B[a]}$$

$$B[a] \to B$$

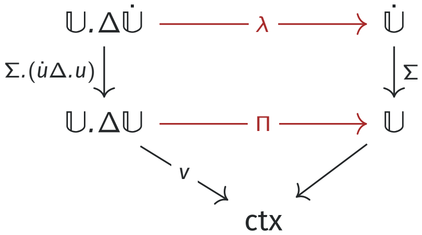
$$\Gamma \to \Gamma.A$$

jDTT, IV: type constructors

Plus, we can define what diagrams one needs to add to DTT in order to get type constructors.

Theorem (3)

It has Π-*types* if it has two additional rules Π, λ such that the diagram below is commutative and the upper square is a pullback.



$$(\Pi I)\ \frac{\Gamma \vdash A \qquad \Gamma.A \vdash b : B}{\Gamma \vdash \lambda_A b : \Pi_A B}$$

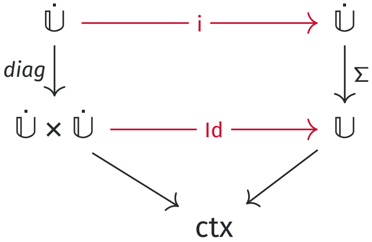
$$(\Pi F)\ \frac{\Gamma \vdash A \qquad \Gamma.A \vdash B}{\Gamma \vdash \Pi_A B}$$

(ΠE): the unique map induced by the pullback from the classifier of (A, B, f, a)
(ΠC) η and β: the two related isomorphisms

jDTT, IV: type constructors

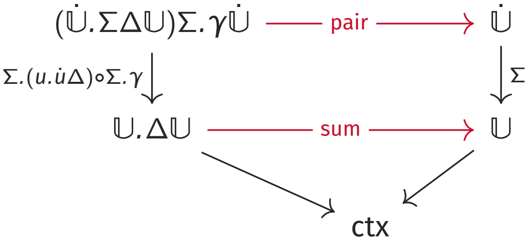
Theorem (4)

It has *extensional* Id-*types* if it has two additional rules Id, i s.t. the diagram below is commutative and the upper square is a pullback.



Theorem (5)

It has *dependent sum types* if it has two additional rules sum, pair s.t. the diagram below is commutative and the upper square is a pullback.



It is clear how one can generalizes this *and* the calculations can be made once for all constructors of this kind.

In summation

We describe a **general theory of judgement** via 2-categorical means and prove its coherence with respect to:

- DTT, and get a (first) general definition of type constructor in the process;
- natural deduction calculus;
- internal logic of a topos.

Still, there are plenty of things that should be looked into, for example:

- prove *some* completeness result;
- extend the theory and the definition to type constructors not included (inductive, coinductive);
- study monads as modalities;
- express new logics (modal? linear?) in this framework.

Thank you for listening! Questions?

Università
di Genova