

Dependent Type Theory

About Extensional Type Systems

Hojune Lee

School of Computing, KAIST

Contents

1	Context and Substitutions	3
1.1	Intuitions	3
1.2	Contexts Judgements	5
1.3	Substitution Judgements	5

Chapter 1

Context and Substitutions

1.1 Intuitions

In simple type theory, every thing was quite clear. This is because that, if we once fix the rules for types in simple type theory, then all possible types are determined immediately by inductive way. We can informally think this as ‘propositional logic’-like type construction. However, many things are different in dependent type system. Let’s think dependent type system as ‘first order logic’-like type construction. Imagine the formulas in first order logic as types in dependent type theory. I hope that this is proper intuitive thinking for the main differences between two type systems.

Once we imagine FOL formula-style type, it means that each type can contain variables and constants as it’s representation. It means that for each variables which are appeared in type representation, we must know what are the types of each variables. That is Context’s role.

Now, we need to construct the intuition for contexts and substitutions. Contexts are ‘worlds’ of terms and types. And substitutions are ‘traffic way’ between each worlds(contexts). Then naturally, following questions are arising.

Questions

1. Is this world (Context) well-formed ?
2. In this world (Context) Γ , what terms of type A are well-formed ?
3. In this world (Context) Γ , what types are well-formed ?
4. Between 2 worlds (Context) Δ and Γ , is traffic way (substitution) γ well-formed ?
5. In this world (Context), which terms/types/substitutions are equivalent?

Actually, this is all about ‘judgement rules’ referred in Notation 2.3.1. of textbook. Formally, we can write format for above judgements.

Formal Representations for Judgements

1. $\vdash \Gamma \text{ cx}$
2. $\Gamma \vdash a : A$
3. $\Gamma \vdash A \text{ type}$
4. $\Delta \vdash \gamma : \Gamma$
5. $\Gamma \vdash a = a' : A, \quad \Gamma \vdash A = A' \text{ type}, \quad \Delta \vdash \gamma = \gamma' : \Gamma$

In this chapter, we'll discuss the rules for above judgements. Now, it's time to define explicit judgement rules that we referred.

1.2 Contexts Judgements

Construction 1.2.1.

$$\frac{}{\vdash \mathbf{1} \text{ cx}} \qquad \frac{\vdash \Gamma \text{ cx} \quad \Gamma \vdash A \text{ type}}{\vdash \Gamma.A \text{ cx}}$$

Here, $\mathbf{1}$ is empty context and each context is just list of types. (No variable names) It means that we use De Bruijn index, i.e. in each context, index automatically determines the variable in context.

1.3 Substitution Judgements

However, before that the most important one is understanding direction of traffic way (from now on, substitution).

Definition 1.3.1.

If $\Delta \vdash \gamma : \Gamma$, then γ is substitution from Δ to Γ . i.e.

$$\gamma : \Delta \rightarrow \Gamma$$

However, it's role is send types and terms of Γ into Δ . (Note : Direction is important)

1.3. Intuitions for direction of substitutions

To understand why here use this notation, see following example. Imagine that contexts are 'sets' and substitutions are function between them. There are 2 sets Δ, Γ and mapping $\gamma : \Delta \rightarrow \Gamma$. Suppose that there is a function $g : \Gamma \rightarrow \mathbb{R}$, which is defined on set Γ . How can we 'use' this function in Δ set? One way is that,

$$g : \Gamma \rightarrow \mathbb{R} \implies g \circ \gamma : \Delta \rightarrow \mathbb{R}$$

Then we can 'use' the function g in domain Δ . This exactly corresponds in our notation. For $\Delta \vdash \gamma : \Gamma$, when we define $\gamma : \Delta \rightarrow \Gamma$,

$$\Gamma \vdash A \text{ type} \implies \Delta \vdash A[\gamma] \text{ type}$$

Since A is type in Γ and the direction of substitution is $\gamma : \Delta \rightarrow \Gamma$, we say that $A[\gamma]$ is pull-backed type of A through $\gamma : \Delta \rightarrow \Gamma$.

First, as above we can easily define the application rule of substitution.

Construction 1.3.2.

$$\frac{\Delta \vdash \gamma : \Gamma \quad \Gamma \vdash A \text{ type}}{\Delta \vdash A[\gamma] \text{ type}} \qquad \frac{\Delta \vdash \gamma : \Gamma \quad \Gamma \vdash a : A}{\Delta \vdash a[\gamma] : A[\gamma]}$$

These rules give us that we can immigrate(pull-back) terms and types of Γ into Δ . And now, in our setting each contexts are different ‘worlds’. So we need to introduce explicit **weakening rule**, which intuitively means that we can bring well-formed types and terms into Γ into $\Gamma.A$, expanded context.

Construction 1.3.3.

$$\frac{\Gamma \vdash A \text{ type}}{\Gamma.A \vdash \mathbf{p} : \Gamma}$$

In diagram,

$$\Gamma.A \xrightarrow{p} \Gamma$$

Means that, we can pull-back types B and terms t in world Γ into $\Gamma.A$ by write $B[\mathbf{p}], t[\mathbf{p}]$. Moreover, we can introduce following rules. (Actually, following rules need because our context-substitution system become a category.)

Construction 1.3.4.

$$\frac{\vdash \Gamma \text{ cx}}{\Gamma \vdash \mathbf{id} : \Gamma} \qquad \frac{\Gamma_2 \vdash \gamma_1 : \Gamma_1 \quad \Gamma_1 \vdash \gamma_0 : \Gamma_0}{\Gamma_2 \vdash \gamma_0 \circ \gamma_1 : \Gamma_0}$$

Second one can be conflict when we see first. However, let’s draw the diagram.

$$\begin{array}{ccc} \Gamma_2 & \xrightarrow{\gamma_1} & \Gamma_1 \\ \gamma_0 \circ \gamma_1 \downarrow & \swarrow \gamma_0 & \\ \Gamma_0 & & \end{array}$$

Then, above representation is very clear. It become same notation in our function calculus. Similarly,

Construction 1.3.5.

$$\frac{\Delta \vdash \gamma : \Gamma}{\Delta \vdash \gamma \circ \mathbf{id} = \mathbf{id} \circ \gamma = \gamma : \Gamma} \qquad \frac{\Gamma_3 \vdash \gamma_2 : \Gamma_2 \quad \Gamma_2 \vdash \gamma_1 : \Gamma_1 \quad \Gamma_1 \vdash \gamma_0 : \Gamma_0}{\Gamma_3 \vdash (\gamma_0 \circ \gamma_1) \circ \gamma_2 = \gamma_0 \circ (\gamma_1 \circ \gamma_2) : \Gamma_0}$$

This two rules are also very clear when we see the diagram.

$$id \hookrightarrow \Delta \xrightarrow{\gamma} \Gamma \hookrightarrow id$$

$$\begin{array}{ccc}
\Gamma_3 & \xrightarrow{\gamma_2} & \Gamma_2 \\
(\gamma_0 \circ \gamma_1) \circ \gamma_2 \downarrow & \swarrow \gamma_0 \circ \gamma_1 & \downarrow \gamma_1 \\
\Gamma_0 & \xleftarrow{\gamma_0} & \Gamma_1
\end{array}
\iff
\begin{array}{ccc}
\Gamma_3 & \xrightarrow{\gamma_2} & \Gamma_2 \\
\gamma_0 \circ (\gamma_1 \circ \gamma_2) \downarrow & \swarrow \gamma_1 \circ \gamma_2 & \downarrow \gamma_1 \\
\Gamma_0 & \xleftarrow{\gamma_0} & \Gamma_1
\end{array}$$

Then we can imagine following equivalent rules clearly.

Construction 1.3.6.

$$\frac{\Gamma \vdash A \text{ type}}{\Gamma \vdash A[\mathbf{id}] = A \text{ type}} \qquad \frac{\Gamma \vdash a : A}{\Gamma \vdash a[\mathbf{id}] = a : A}$$

$$\frac{\Gamma_2 \vdash \gamma_1 : \Gamma_1 \quad \Gamma_1 \vdash \gamma_0 : \Gamma_0 \quad \Gamma_0 \vdash A \text{ type}}{\Gamma_2 \vdash A[\gamma_0 \circ \gamma_1] = A[\gamma_0][\gamma_1] \text{ type}}$$

$$\frac{\Gamma_2 \vdash \gamma_1 : \Gamma_1 \quad \Gamma_1 \vdash \gamma_0 : \Gamma_0 \quad \Gamma_0 \vdash a : A}{\Gamma_2 \vdash a[\gamma_0 \circ \gamma_1] = a[\gamma_0][\gamma_1] : A[\gamma_0 \circ \gamma_1]}$$

These rules are very intuitable. When we draw diagram,

$$\begin{array}{ccc}
\Gamma_2 & \xrightarrow{\gamma_1} & \Gamma_1 \\
\gamma_0 \circ \gamma_1 \downarrow & \swarrow \gamma_0 & \\
\Gamma_0 & &
\end{array}$$

For example, look the last rule. a is term in the world Γ_0 . We want to bring this term into Γ_2 . Bring means that, we want to use ‘function’ a in Γ_2 . i.e. such term in Γ_2 is working same roles with a in Γ_0 . We’ve studied in box 1.3, there are 2 ways to bring a from Γ_0 into Γ_2 .

$$\begin{array}{ccc}
\Gamma_2 & \xrightarrow{\gamma_1} & \Gamma_1 \\
\gamma_0 \circ \gamma_1 \downarrow & \swarrow \gamma_0 & \\
\Gamma_0 & &
\end{array}$$

One is pulling back via red way, and another one is via blue way. a via red way is $a[\gamma_0 \circ \gamma_1]$ in Γ_2 , and via blue way is $a[\gamma_0][\gamma_1]$ (Actually, $a[\gamma_0]$ is a in Γ_1 world. So we take it again into Γ_2 .) However, how can we use variable in well-formed context Γ without own names? What is the meaning of De Bruijn index? Let’s see how can we distinguish variables in each ‘world’ Γ .

Construction 1.3.7.

$$\frac{\Gamma \vdash A \text{ type}}{\Gamma.A \vdash \mathbf{q} : A[\mathbf{p}]}$$

We can see context as a stack. \mathbf{q} always means the top element of stack. However, stack $\Gamma.A$ contains type A as top, which is type in Γ world. So the top variable's type in $\Gamma.A$ is $A[\mathbf{p}]$ which we take A from Γ into $\Gamma.A$. Similarly, we can access the any index of stack by following rule which is derivable from 1.3

Corolary 1.3.8.

$$\frac{\Gamma \vdash A \text{ type} \quad \Gamma.A \vdash B_1 \text{ type} \quad \cdots \Gamma.A.B_1 \cdots \vdash B_n \text{ type}}{\Gamma.A.B_1 \cdots .B_n \vdash \mathbf{q}[\mathbf{p}^n] : A[\mathbf{p}^{n+1}]}$$

Now, it's time to discuss our main theme in this subsection 1.3. Our ultimate goal is defining 'valid traffic way from valid world to another valid world'. Let's discuss substitution judgement rules very intuitively. First, suppose that we only have rules for context judgements 1.2. The obvious thing that we must do first is, construct way from any Γ into $\mathbf{1}$. The rules are simple.

Construction 1.3.9.

$$\frac{\vdash \Gamma \text{ ex}}{\Gamma \vdash ! : \mathbf{1}} \qquad \frac{\Gamma \vdash \delta : \mathbf{1}}{\Gamma \vdash \delta =! : \mathbf{1}}$$

What does it mean? It is 'way' from Γ to $\mathbf{1}$. The role of this way is transfer terms and types in $\mathbf{1}$ into Γ world. What terms and types are well-formed in $\mathbf{1}$? Just closed terms and types (No Variables). It is obviously terms and types in Γ also. So we define the way is 'unique' here. With this substitution, we can always transfer closed terms (e.g. $1, (1+1)$) and closed types from $\mathbf{1}$ to any well-formed context Γ . Let's think this as a base case of our definition of traffic ways. Following construction determines how can we extends substitutions.

Construction 1.3.10.

$$\frac{\Delta \vdash \gamma : \Gamma \quad \Gamma \vdash A \text{ type} \quad \Delta \vdash a : A[\gamma]}{\Delta \vdash \gamma.a : \Gamma.A}$$

Let's draw diagram.

$$\begin{array}{ccc} \Delta & \xrightarrow{\gamma} & \Gamma \\ & \searrow \gamma.a & \uparrow p \downarrow id.? \\ & & \Gamma.A \end{array}$$

Is it clear? It means that, suppose that Γ has n variables (x_1, \dots, x_n) . Here we denote names for each variables for representation. And then, in this context Γ , A is well-formed type. Then automatically $\Gamma.A$ is defined, where it has $n + 1$ variables $(x_1, \dots, x_n, x_{n+1})$.

The key understanding is that, when we bring terms and types in $\Gamma.A$ world, it can have x_1, \dots, x_{n+1} as symbol. When we meet x_{n+1} , then substitution it by a , and when we meet x_1, \dots, x_n , then substitution it by following γ . Then since a is well-formed term in Δ and it's type is $A[\gamma]$, which do roles of A in Δ world, so such substitution also works well. This is very powerful intuition. Now, with above understanding, let's define more challenging and powerful rules for substitutions.

Construction 1.3.11.

$$\frac{\Delta \vdash \gamma : \Gamma \quad \Gamma \vdash A \text{ type} \quad \Delta \vdash a : A[\gamma]}{\Delta \vdash \mathbf{p} \circ (\gamma.a) = \gamma : \Gamma}$$

$$\frac{\Delta \vdash \gamma : \Gamma \quad \Gamma \vdash A \text{ type} \quad \Delta \vdash a : A[\gamma]}{\Delta \vdash \mathbf{q}[\gamma.a] = a : A[\gamma]}$$

$$\frac{\Gamma \vdash A \text{ type} \quad \Delta \vdash \gamma : \Gamma.A}{\Delta \vdash \gamma = (\mathbf{p} \circ \gamma).\mathbf{q}[\gamma] : \Gamma.A}$$

There rules are seemed to be very unclear. However, see following diagram. For first rule,

$$\begin{array}{ccc} \Delta & \xrightarrow{\gamma} & \Gamma \\ & \searrow \gamma.a & \uparrow p \downarrow id.? \\ & & \Gamma.A \end{array}$$

We can represent the first rule as following way.

$$\begin{array}{ccc} \Delta & \xrightarrow{\gamma} & \Gamma \\ & \searrow \gamma.a & \uparrow p \downarrow id.? \\ & & \Gamma.A \end{array}$$

So we can intuitively know that γ and $\mathbf{p} \circ (\gamma.a)$ represent same traffic way from Δ to Γ . For second rule, we already explained in front page. When we meet x_{n+1} (i.e. meet q in $\Gamma.A$) then bring it from $\Gamma.A$ into Δ and substitute by a , term of Δ . This rule explicitly claim this intuition. For third rule, let's draw slightly changed diagram.

$$\begin{array}{ccc} \Delta & & \Gamma \\ & \searrow \gamma & \uparrow p \downarrow id.? \\ & & \Gamma.A \end{array}$$

Can we see what is $(\mathbf{p} \circ \gamma)$ here? We can draw

$$\begin{array}{ccc} \Delta & \xrightarrow{p \circ \gamma} & \Gamma \\ & \searrow \gamma & \uparrow p \downarrow id.? \\ & & \Gamma.A \end{array}$$

Now, see 1.3. When we make $\mathbf{p} \circ \gamma : \Delta \rightarrow \Gamma$ to something that $\Delta \rightarrow \Gamma.A$, we append something to $\mathbf{p} \circ \gamma$. Here, actually the answer is already given. Suppose q in $\Gamma.A$, it has type $A[\mathbf{p}]$ in $\Gamma.A$. We define γ by, how we take variable \mathbf{q} in $\Gamma.A$ to Δ ? Actually its answer is $\mathbf{q}[\gamma]$. It has type $A[\mathbf{p}][\gamma] = A[\mathbf{p} \circ \gamma]$ in Δ . So to make same substitution with γ , we must append $\mathbf{q}[\gamma]$ at $\mathbf{p} \circ \gamma$. This intuition gives us $\gamma = (\mathbf{p} \circ \gamma).\mathbf{q}[\gamma]$ in Δ .

.