The Biequivalence of Locally Cartesian Closed Categories and Martin-Löf Type Theory

Pierre Clairambault, Bath
and
Peter Dybjer, Chalmers

TLCA 2011
Categorical logic: key correspondences

Cartesian closed categories $\simeq$ simply typed $\lambda$-calculus

Hyperdoctrines $\simeq$ first-order logic

Toposes $\simeq$ Higher-order logic

? $\simeq$ Martin-Löf type theory
R. Seely (1984), Locally cartesian closed categories and type theory:

6.3. **THEOREM.** The categories **ML** and **LCC** are equivalent.

- **ML** is the category of "Martin-Löf theories" with types $\prod_{x \in A} B[x]$, $\sum_{x \in A} B[x]$, and $l(a, b)$.
- **LCC** is the category of locally cartesian closed categories.
We consider *extensional* intuitionistic type theory of Martin-Löf (1979, 1984), i.e. identity types satisfy:

\[
\begin{align*}
\vdash p : I_A(m, n) & \quad \vdash p : I_A(m, n) \\
\vdash m = n : A & \quad \vdash p = r_A(m) : I_A(m, n)
\end{align*}
\]

Note that these rules are *not* derivable in intensional type theory (and break decidability of type checking).
A category $C$ is **locally cartesian closed (lccc)** iff either of the following equivalent conditions hold:

- All slice categories $C/A$ are cartesian closed (and $C$ has a terminal object),
- $C$ has finite limits and the functor $f^* : C/B \to C/A$ has a right adjoint $\Pi_f$ for $f : A \to B$. (The left adjoint $\Sigma_f$ always exists.)

Seely’s **LCC** is the category of lcccs and lccc-structure preserving functors.
Martin-Löf theory and lccc - correspondences

- Contexts are objects of $C$.
- Types in context $\Gamma$ are morphisms $A : dom(A) \to \Gamma$ (objects of $C/\Gamma$), called display maps,
- Terms of type $A$ are sections of $A$, i.e. morphisms $a : \Gamma \to dom(A)$ such that $A \circ a = id_\Gamma$.
- Type substitution is pullback:

$$
\begin{array}{ccc}
\Delta & \xrightarrow{f} & \Gamma \\
\downarrow & & \downarrow \\
 f^*A & \xrightarrow{A} & A \\
\end{array}
$$

- Extensional identity types are equalizers
- $\Sigma$-types are left adjoints $\Sigma_f \dashv f^*$
- $\Pi$-types are right adjoints $f^* \dashv \Pi_f$
Flaw in Seely’s proof: pullbacks compose only up to isomorphism.

\[ A[M/x][N/y] = A[M[N/y]/x] \]

\[
\begin{array}{ccc}
[D] & \xrightarrow{[N]} & C \\
 & \downarrow{[M]} & \\
 & [M] & \xrightarrow{[A]} \\
 & \downarrow{[N]} & \\
D & \xrightarrow{[M]} & B \\
& \downarrow{[M]\circ[N]} & \\
& D & \xrightarrow{[A]} \\
& \downarrow & \\
& B & \\
\end{array}
\]

It amounts to the fact that a locally cartesian closed category, as a fibration, is not necessarily split.
Curien and Hofmann

Two proposed solutions for the interpretation of MLTT $\mathcal{T}$ in a LCCC $C$:

\[
\begin{align*}
\text{Curien (93)} &: \quad \mathcal{T}^* \rightarrow C \\
\text{Hofmann (95)} &: \quad \mathcal{T} \rightarrow \mathcal{B}(C)
\end{align*}
\]

Where:

- $\mathcal{T}^*$ is an extension of $\mathcal{T}$ with \textit{explicit substitutions}, and special terms for \textit{type isomorphisms}.
- $\mathcal{B}$ is the \textit{Bénabou construction}, which associates to any fibration an equivalent \textit{split} fibration.

However only the interpretation is investigated, not whether this gives an \textit{equivalence}. 
A type $A$ over $\Gamma$ is no longer a *display map* $A : \text{dom}(A) \to \Gamma$, but a *family* $\vec{A}$ of display maps:

\[
\begin{array}{ccc}
\downarrow \vec{A}(\delta \gamma) & \downarrow \vec{A}(\delta) & \downarrow \vec{A}(\text{id}_\Gamma) \\
\Omega \gamma & \Delta \delta & \Gamma
\end{array}
\]

Such that all the squares are pullback squares.

**Definition**

A *functorial family* over $\Gamma$ is a functor $\vec{A} : C/\Gamma \to C^\to$ satisfying some conditions.
Our contribution

Seely’s statement

*The categories LCC and ML are equivalent.*

Theorem

*The 2-categories LCC and CwF\textsubscript{\text{I\text{ext}, \Sigma, \Pi and dem}} are biequivalent.*

- Instead of an equivalence, we have a weaker *biequivalence*.
- We use *categories with families (CwF)* with extra structure (\text{I\text{ext}, \Sigma, \Pi and dem}) as a replacement for syntax.
What is a biequivalence?

When are two objects "abstractly the same"?

**Equality**

(set) \( a = b \)

**Isomorphism**

(category) \( a \cong b \)

\[
\begin{array}{ccc}
& f & \\
& \downarrow & \\
& b & \leftarrow g & a \\
& \uparrow & \\
& b & \rightarrow f & a
\end{array}
\]

\( fg = 1_b \)
\( gf = 1_a \)

**Equivalence**

(bicategory) \( a \simeq b \)

\[
\begin{array}{ccc}
& f & \\
& \downarrow & \\
& b & \leftarrow g & a \\
& \uparrow & \\
& b & \rightarrow f & a
\end{array}
\]

\( fg \cong 1_b \)
\( gf \cong 1_a \)

**Biequivalence**

(tricategory) \( a \sim b \)

\[
\begin{array}{ccc}
& f & \\
& \downarrow & \\
& b & \leftarrow g & a \\
& \uparrow & \\
& b & \rightarrow f & a
\end{array}
\]

\( gf \simeq 1_a \)
\( fg \simeq 1_b \)
What is a biequivalence?

To define biequivalent 2-categories, we consider the tricategory of bicategories which components are:

- **0-cells**: bicategories
- **1-cells**: pseudofunctors between bicategories

```
\[
\begin{array}{ccc}
FA & \xrightarrow{\phi_A} & FA \\
\downarrow^{-} & & \downarrow^{-} \\
F1_A & & F1_A \\
\end{array}
\quad
\begin{array}{ccc}
FA & \xrightarrow{F(f)} & FB \\
\downarrow{\phi_{f,g}} & & \downarrow{\phi_{f,g}} \\
FA & \xrightarrow{F(g)} & FC \\
\end{array}
\]
```

- **2-cells**: pseudonatural transformation of pseudofunctors

```
\[
\begin{array}{ccc}
FA & \xrightarrow{\eta_A} & GA \\
\downarrow{Ff} & & \downarrow{Gf} \\
FB & \xrightarrow{\eta_B} & GB \\
\end{array}
\]
```

- **3-cells**: modification of pseudonatural transformations
Proving the biequivalence

We need to provide the following data (and check the appropriate properties):

- **LCC**: the 2-category of locally cartesian closed categories, structure-preserving functors, and natural transformations.
- **CwF\textsubscript{\text{I}ext\Sigma\Pi}^{\text{dem}}**: the 2-category of cwfs with extra structure.
- **U**: \(CwF\textsubscript{\text{I}ext\Sigma\Pi}^{\text{dem}} \rightarrow \text{LCC}\) is a forgetful 2-functor.
- **H**: \(\text{LCC} \rightarrow CwF\textsubscript{\text{I}ext\Sigma\Pi}^{\text{dem}}\) is a pseudofunctor based on the Bénabou-Hofmann construction.
- \(\eta: 1 \rightarrow HU\) and \(\epsilon: HU \rightarrow 1\): pseudonatural transformations, which are inverses up to invertible modifications \(\phi, \psi\).
A category with family \((C, T)\) is the data of:

- \(C\), a category of contexts.
- \(T : C^{op} \to \text{Fam}\), a functor where the
  - object part \(\Gamma \mapsto (\{ a \mid \Gamma \vdash a : A \})_{A \in \text{Type}(\Gamma)}\)
  - arrow part \(\gamma \mapsto \begin{cases} A \mapsto A[\gamma] \\ a \mapsto a[\gamma] \end{cases}\)

- A terminal object [] of \(C\) called the empty context.
- A context comprehension operation which to an object \(\Gamma\) of \(C\) and a type \(A \in \text{Type}(\Gamma)\) associates a context \(\Gamma \cdot A\) satisfying a product-like universal property.

Cwfs can be presented as a generalised algebraic theory, and be seen as a variable-free syntax for Martin-Löf type theory with explicit substitutions.
Cwf-morphisms

**Definition**

A *strict cwf-morphism* from \((C, T)\) to \((\mathcal{D}, T')\) is a pair:
- \(F : C \to \mathcal{D}\) is a functor;
- \(\sigma : T \to T'F\) is a natural transformation.

It follows that \(\sigma_\Gamma(A)[F\delta] = \sigma_\Delta(A[\delta])\)

**Definition**

A *pseudo cwf-morphism* from \((C, T)\) to \((\mathcal{D}, T')\) is a pair:
- \(F : C \to \mathcal{D}\) is a functor;
- \(\sigma_\Gamma : T\Gamma \to T'F\Gamma\) is a family of Fam-morphism.

Such that \(\sigma_\Gamma(A)[F\delta] \cong \sigma_\Delta(A[\delta])\), with coherence conditions.
The 2-category \( \text{CwF}^{I_{\text{ext}} \Sigma \Pi}_{\text{dem}} \)

**Definition**

There is a 2-category \( \text{CwF}^{I_{\text{ext}} \Sigma \Pi}_{\text{dem}} \) with:

- 0-cells are cwfs supporting \( \Sigma, \Pi, I_{\text{ext}} \) and \( \text{dem} \),
- 1-cells are pseudo cwf-morphisms which also preserve \( \Sigma, \Pi, I_{\text{ext}} \) and \( \text{dem} \) up to isomorphism,
- 2-cells are pseudo cwf-transformations.
The remaining components

- \( U(C, T) = C \) is a forgetful 2-functor,

- \( H(C) = (C, T_C) \) generalises the Bénabou-Hofmann construction to a pseudofunctor.
The remaining components

- $\eta_{(C,T)} = (Id, \sigma)$ where $\sigma(A)$ associates to any substitution $\delta$ the display map of $A[\delta]$.

- $\epsilon_{(C,T)} = (Id, \tau)$ where $\tau(\overrightarrow{A}) = \Sigma_{y: \Delta} l_{\Gamma}(\overrightarrow{A}(id)(y), x)(x : \Gamma)$ builds a syntactic representative type for $\overrightarrow{A}$. 
## Conclusion

**Theorem**

The 2-categories $\textbf{LCC}$ and $\textbf{CwF}_{\text{ext} \Sigma \Pi}^{\text{dem}}$ are biequivalent.

Removing everything related to $\Pi$ in the proof yields the following result:

**Theorem**

The 2-categories $\textbf{FL}$ and $\textbf{CwF}_{\text{ext} \Sigma}^{\text{dem}}$ are biequivalent.