Introducing a New Project on The Combination of Substructural Logics and Dependent Type Theory

Harley Eades III
Computer Science
Augusta University

Proof Assistants have been extremely useful for formalizing large scale resource dependent systems

CompCert

seL4

But, there is a limitation!

"…managing assumption contexts does not work for the substructural separation logic and therefore needs to be done manually."

Gerwin Klein et al. "Mechanised separation algebra."

I propose that we mix substructural logics with dependent types at the foundational level, rather than, as an add on.

Two Main Problems

- How to modularly support many different substructural logics?
- How to integrate the substructural logical framework with dependent types?

A Basic Substructural Logic

A magmoidal category with a unit has the following data:

- $\odot : \mathcal{M} \times \mathcal{M} \to \mathcal{M}$
- $I \in \mathsf{Obj}(\mathcal{M})$
- $\lambda_A : A \odot I \to A$
- $\rho_A : I \odot A \to A$

The Five Basic Substructural Logics

- $\mathcal{M}$: None
- $\mathcal{A}$: Associative
- $\mathcal{E}$: Commutative
- $\mathcal{W}$: Affine
- $\mathcal{C}$: Contractive

Adjoint Models

$$\begin{tikzcd} \mathcal{S}_2 \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\perp"] & & \mathcal{S}_1 \arrow[ll, bend left, "G"] \end{tikzcd}$$

Composition of Substructural Logics

$$\begin{tikzcd} \mathcal{S}_3 \arrow[r, bend left, "F_2"] \arrow[r, phantom, "\perp"] & \mathcal{S}_2 \arrow[l, bend left, "G_2"] \end{tikzcd} \qquad \begin{tikzcd} \mathcal{S}_2 \arrow[r, bend left, "F_1"] \arrow[r, phantom, "\perp"] & \mathcal{S}_1 \arrow[l, bend left, "G_1"] \end{tikzcd}$$

Composition of Substructural Logics

$$\begin{tikzcd} \mathcal{S}_3 \arrow[r, bend left, "F_2"] \arrow[r, phantom, "\perp"] & \mathcal{S}_2 \arrow[l, bend left, "G_2"] \arrow[r, bend left, "F_1"] \arrow[r, phantom, "\perp"] & \mathcal{S}_1 \arrow[l, bend left, "G_1"] \end{tikzcd}$$

A Concrete Example

$$
\begin{tikzcd}
\mathcal{W} \arrow[r, bend left, "F_{we}"] \arrow[r, phantom, "\perp"] & \mathcal{E} \arrow[l, bend left, "G_{ew}"] \arrow[r, bend left, "F_{ea}"] \arrow[r, phantom, "\perp"] & \mathcal{A} \arrow[l, bend left, "G_{ae}"]
\end{tikzcd}
$$

$$
\begin{aligned}
A, B, C ::= {} & I_a \\
\mid {} & A \rhd_a B \\
\mid {} & \mathsf{F}_{ea}\, L
\end{aligned}
$$

A Concrete Example

$$\begin{tikzcd} \mathcal{W} \arrow[r, bend left, "F_{we}"] \arrow[r, phantom, "\perp"] & \mathcal{E} \arrow[l, bend left, "G_{ew}"] \arrow[r, bend left, "F_{ea}"] \arrow[r, phantom, "\perp"] & \mathcal{A} \arrow[l, bend left, "G_{ae}"] \end{tikzcd}$$

$$\begin{aligned} L, M, N ::= {} & I_e \\ \mid {} & L \rhd_e M \\ \mid {} & \mathsf{G}_{ae}\, A \\ \mid {} & \mathsf{F}_{we}\, X \end{aligned}$$

A Concrete Example

$$\begin{tikzcd} \mathcal{W} \arrow[r, bend left, "F_{we}"] \arrow[r, phantom, "\perp"] & \mathcal{E} \arrow[l, bend left, "G_{ew}"] \arrow[r, bend left, "F_{ea}"] \arrow[r, phantom, "\perp"] & \mathcal{A} \arrow[l, bend left, "G_{ae}"] \end{tikzcd}$$

$$\begin{aligned} X, Y, Z ::= {} & I_w \\ \mid {} & X \triangleright_w Y \\ \mid {} & \mathsf{G}_{ew}\, L \end{aligned}$$

A Concrete Example

$\boxed{\Phi \vdash_{\mathcal{W}} X}$ $\boxed{\Delta \vdash_{\mathcal{E}} L}$ $\boxed{\Gamma \vdash_{\mathcal{A}} A}$

$$\begin{array}{rcl} \Phi & ::= & \cdot \\ & \mid & X \\ & \mid & \Phi_1 ; \Phi_2 \end{array}$$

$$\begin{array}{rcl} \Delta & ::= & \cdot \\ & \mid & L \\ & \mid & X \\ & \mid & \Delta_1 ; \Delta_2 \end{array}$$

$$\begin{array}{rcl} \Gamma & ::= & \cdot \\ & \mid & A \\ & \mid & L \\ & \mid & X \\ & \mid & \Gamma_1 ; \Gamma_2 \end{array}$$

A Concrete Example

$$\frac{}{A \vdash_{\mathcal{A}} A}\ \mathrm{id} \qquad \frac{}{\cdot \vdash_{\mathcal{A}} I_a}\ I_i$$

A Concrete Example

$$\frac{\Gamma_1 \vdash_{\mathcal{A}} A \quad \Gamma_2 \vdash_{\mathcal{A}} B}{\Gamma_1; \Gamma_2 \vdash_{\mathcal{A}} A \rhd_a B}\, T_i$$

$$\frac{\Gamma_2 \vdash_{\mathcal{A}} A \rhd_a B \quad \Gamma_1; A; B; \Gamma_3 \vdash_{\mathcal{A}} C}{\Gamma_1; \Gamma_2; \Gamma_3 \vdash_{\mathcal{A}} C}\, T_e$$

A Concrete Example

$$\frac{\Delta \vdash_{\mathcal{E}} L}{\Delta \vdash_{\mathcal{A}} \mathsf{F}_{ea}\, L}\, F_i \qquad \frac{\Gamma_2 \vdash_{\mathcal{A}} \mathsf{F}_{ea}\, L \quad \Gamma_1; L; \Gamma_3 \vdash_{\mathcal{A}} A}{\Gamma_1; \Gamma_2; \Gamma_3 \vdash_{\mathcal{A}} A}\, F_e$$

$$\frac{\Delta \vdash_{\mathcal{E}} \mathsf{G}_{ae}\, A}{\Delta \vdash_{\mathcal{A}} A}\, G_e$$

A Concrete Example

$$\frac{\Delta_1; L; M; \Delta_2 \vdash_\varepsilon N}{\Delta_1; M; L; \Delta_2 \vdash_\varepsilon N} E$$

A Concrete Example

$$\frac{\Phi \vdash_{\mathcal{I}} X}{\Phi \vdash_{\mathcal{E}} \mathsf{F}_{we}\, X}\, F_i \qquad \frac{\Delta_2 \vdash_{\mathcal{E}} \mathsf{F}_{we}\, X \quad \Delta_1; X; \Delta_3 \vdash_{\mathcal{E}} L}{\Delta_1; \Delta_2; \Delta_3 \vdash_{\mathcal{E}} L}\, F_e$$

$$\frac{\Phi \vdash_{\mathcal{I}} \mathsf{G}_{ew}\, L}{\Phi \vdash_{\mathcal{E}} L}\, G_e \qquad \frac{\Delta \vdash_{\mathcal{A}} A}{\Delta \vdash_{\mathcal{E}} \mathsf{G}_{ae}\, A}\, G_i$$

A Concrete Example

$$\frac{\Phi_1; \Phi_2 \vdash_{\mathcal{I}} Y}{\Phi_1; X; \Phi_2 \vdash_{\mathcal{I}} Y}\, W \qquad\qquad \frac{\Phi \vdash_{\mathcal{E}} L}{\Phi \vdash_{\mathcal{I}} \mathsf{G}_{ew}\, L}\, G_i$$

A Concrete Example

$$
\begin{tikzcd}
\mathcal{W} \arrow[rr, bend left, "F_{we}"] \arrow[rr, phantom, "\perp"] && \mathcal{E} \arrow[ll, bend left, "G_{ew}"] \arrow[rr, bend left, "F_{ea}"] \arrow[rr, phantom, "\perp"] && \mathcal{A} \arrow[ll, bend left, "G_{ae}"]
\end{tikzcd}
$$

$\mathsf{AF}\,A \vdash_{\mathcal{A}} I_a$ $\qquad$ $\mathsf{W}\,L \vdash_{\mathcal{E}} I_e$

$(\mathsf{AF}\,A) \rhd_a (\mathsf{AF}\,B) \vdash_{\mathcal{A}} (\mathsf{AF}\,B) \rhd_a (\mathsf{AF}\,A)$

$(\mathsf{LIN}\,A) \rhd_a (\mathsf{LIN}\,B) \vdash_{\mathcal{A}} (\mathsf{LIN}\,B) \rhd_a (\mathsf{LIN}\,A)$

Dependent Types

$$\begin{tikzcd} \mathcal{D} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\perp"] & & \mathcal{L} \arrow[ll, bend left, "G"] \end{tikzcd}$$

$$\Gamma \vdash F(e, t) : Fx : X.A$$

See: Krishnaswami, Pradic, and Benton's, “Integrating Linear and Dependent Types”, POPL'15

The Tenli Proof Assistant