Higher-Order Focusing on Linearity and Effects

Siva Somayyajula

6/11/2025

Introduction

Polarization in Focused and Effect Calculi

Introduction

Polarization in Focused and Effect Calculi

- Focused and effect calculi share **polarization** of types

Introduction

Polarization in Focused and Effect Calculi

- Focused and effect calculi share **polarization** of types
- In call-by-push-value and friends: **value** and **computation** types

Introduction

Polarization in Focused and Effect Calculi

- Focused and effect calculi share **polarization** of types
- In call-by-push-value and friends: **value** and **computation** types
- Focusing has **positive** and **negative** formulae

Introduction

Polarization in Focused and Effect Calculi

- Focused and effect calculi share **polarization** of types
- In call-by-push-value and friends: **value** and **computation** types
- Focusing has **positive** and **negative** formulae
- This talk:

Introduction

Polarization in Focused and Effect Calculi

- Focused and effect calculi share **polarization** of types
- In call-by-push-value and friends: **value** and **computation** types
- Focusing has **positive** and **negative** formulae
- This talk:
  - Existing correspondence, e.g., between focused intuitionistic logic and call-by-push-value (CBPV)

Introduction

Polarization in Focused and Effect Calculi

- Focused and effect calculi share **polarization** of types
- In call-by-push-value and friends: **value** and **computation** types
- Focusing has **positive** and **negative** formulae
- This talk:
  - Existing correspondence, e.g., between focused intuitionistic logic and call-by-push-value (CBPV)
  - A **higher-order** focused analogue of the enriched effect calculus (EEC)

Focusing

A Quick Introduction

Focusing

A Quick Introduction

- Distinguish formulas (A) by **polarity**: **positive** (P, Q) and **negative** (N, O).

Focusing

A Quick Introduction

- Distinguish formulas (A) by **polarity**: **positive** (P, Q) and **negative** (N, O).
- Intuitionistic sequent calculus: **non-backtracking** left rules (+) vs. right rules (-)

Focusing

A Quick Introduction

- Distinguish formulas (A) by **polarity**: **positive** (P, Q) and **negative** (N, O).
- Intuitionistic sequent calculus: **non-backtracking** left rules (+) vs. right rules (-)

$$\Gamma \vdash A$$

Focusing

A Quick Introduction

- Distinguish formulas (A) by **polarity**: **positive** (P, Q) and **negative** (N, O).
- Intuitionistic sequent calculus: **non-backtracking** left rules (+) vs. right rules (-)

$$\Gamma \vdash A \quad \longrightarrow$$

Focusing

A Quick Introduction

- Distinguish formulas (A) by **polarity**: **positive** (P, Q) and **negative** (N, O).
- Intuitionistic sequent calculus: **non-backtracking** left rules (+) vs. right rules (-)

$$\Gamma^-; \Gamma^+ \vdash A$$ inversion (non-backtracking rules)

$$\Gamma \vdash A \quad \Rightarrow$$

Focusing

A Quick Introduction

- Distinguish formulas (A) by **polarity**: **positive** (P, Q) and **negative** (N, O).
- Intuitionistic sequent calculus: **non-backtracking** left rules (+) vs. right rules (-)

$$\Gamma^-; \Gamma^+ \vdash A$$ inversion (non-backtracking rules)

$$\Gamma \vdash A \quad \Longrightarrow \quad \Gamma^- \vdash [P]$$ right focus (backtracking right rules)

Focusing

A Quick Introduction

- Distinguish formulas (A) by **polarity**: **positive** (P, Q) and **negative** (N, O).
- Intuitionistic sequent calculus: **non-backtracking** left rules (+) vs. right rules (-)

$$\Gamma^-; \Gamma^+ \vdash A$$ inversion (non-backtracking rules)

$$\Gamma \vdash A \quad \Longrightarrow \quad \Gamma^- \vdash [P]$$ right focus (backtracking right rules)

$$\Gamma^- \vdash N > P$$ left focus (backtracking left rules)

Focusing

A Quick Introduction

- Distinguish formulas (A) by **polarity**: **positive** (P, Q) and **negative** (N, O).
- Intuitionistic sequent calculus: **non-backtracking** left rules (+) vs. right rules (-)

$$\Gamma^-; \Gamma^+ \vdash A$$ inversion (non-backtracking rules)

$$\Gamma \vdash A \quad \Longrightarrow \quad \Gamma^- \vdash [P]$$ right focus (backtracking right rules)

$$\Gamma^- \vdash N > P$$ left focus (backtracking left rules)

- **Focalization**: complete for classical linear logic (Andreoli, 1992), intuitionistic logic (Liang & Miller, 2009), etc.

A Correspondence

Between Types, Terms, and Reduction

A Correspondence

Between Types, Terms, and Reduction

	focused intuitionistic logic	call-by-push-value

A Correspondence

Between Types, Terms, and Reduction

	focused intuitionistic logic	**call-by-push-value**
types	positive	value

A Correspondence

Between Types, Terms, and Reduction

	focused intuitionistic logic	call-by-push-value
types	positive	value
	negative	computation

A Correspondence

Between Types, Terms, and Reduction

	focused intuitionistic logic	call-by-push-value
types	positive	value
	negative	computation
terms*	inversion	computation

A Correspondence

Between Types, Terms, and Reduction

	focused intuitionistic logic	call-by-push-value
types	positive	value
	negative	computation
terms*	inversion	computation

*opposing contextual structure (insightful comments on Krishnaswami, 2018)

A Correspondence

Between Types, Terms, and Reduction

	focused intuitionistic logic	call-by-push-value
types	positive	value
	negative	computation
terms*	inversion	computation
	right focus	value

*opposing contextual structure (insightful comments on Krishnaswami, 2018)

A Correspondence

Between Types, Terms, and Reduction

	focused intuitionistic logic	call-by-push-value
types	positive	value
	negative	computation
terms*	inversion	computation
	right focus	value
	weak left focus	**stack**

*opposing contextual structure (insightful comments on Krishnaswami, 2018)

A Correspondence

Between Types, Terms, and Reduction

	focused intuitionistic logic	call-by-push-value
types	positive	value
	negative	computation
terms*	inversion	computation
	right focus	value
	weak left focus	**stack**

The CK-machine semantics for CBPV plays a computation of type N off a stack typed by the judgment $\Gamma^{+} \vdash N \gg O$, producing a result computation of type O

*opposing contextual structure (insightful comments on Krishnaswami, 2018)

A Correspondence

Between Types, Terms, and Reduction

	focused intuitionistic logic	call-by-push-value
types	positive	value
	negative	computation
terms*	inversion	computation
	right focus	value
	weak left focus	**stack**
reduction	focalization of CBPV term is βη-equivalent to it (Rioux and Zdancewic, 2020)	

The CK-machine semantics for CBPV plays a computation of type N off a stack typed by the judgment $\Gamma^+ \vdash N \gg O$, producing a result computation of type O

*opposing contextual structure (insightful comments on Krishnaswami, 2018)

A Correspondence

Between Calculi

A Correspondence

Between Calculi

focusing	effect calculus

A Correspondence

Between Calculi

focusing	effect calculus
(weakly) focused intuitionistic logic	call-by-push-value + stacks

A Correspondence

Between Calculi

focusing	effect calculus
(weakly) focused intuitionistic logic	call-by-push-value + stacks
? (this talk, kind of)	**enriched effect calculus (EEC)**

A Correspondence

Between Calculi

focusing	effect calculus
(weakly) focused intuitionistic logic	call-by-push-value + stacks
? (this talk, kind of)	**enriched effect calculus (EEC)**

Issue: EEC disagrees with focusing-theoretic polarity

A Correspondence

Between Calculi

focusing	effect calculus
(weakly) focused intuitionistic logic	call-by-push-value + stacks
? (this talk, kind of)	**enriched effect calculus (EEC)**
focused intuitionistic linear logic	linear L-calculi (Curien et al., 2016; subsumes EEC)

Issue: EEC disagrees with focusing-theoretic polarity

EEC

Another Quick Introduction

EEC

Another Quick Introduction

- Observation: a stack constitutes a *linear* consumption of the input computation

EEC

Another Quick Introduction

- Observation: a stack constitutes a *linear* consumption of the input computation

- Idea: **internalize** stack judgment $N \gg O$ as the **value type** $N \multimap O$

EEC

Another Quick Introduction

- Observation: a stack constitutes a *linear* consumption of the input computation
- Idea: **internalize** stack judgment $N \gg O$ as the **value type** $N \multimap O$
  - Left rule internalizes CK-semantics, “looks like” function application

EEC

Another Quick Introduction

- Observation: a stack constitutes a *linear* consumption of the input computation
- Idea: **internalize** stack judgment $N \gg O$ as the **value type** $N \multimap O$
    - Left rule internalizes CK-semantics, “looks like” function application
- Problem: function types *shouldn’t* be positive (?)

EEC

Another Quick Introduction

- Observation: a stack constitutes a *linear* consumption of the input computation
- Idea: **internalize** stack judgment $N \gg O$ as the **value type** $N \multimap O$
  - Left rule internalizes CK-semantics, “looks like” function application
- Problem: function types *shouldn’t* be positive (?)
- Other EEC connectives have similar problems ⇒ different notion of polarity?

Toward a Solution

Higher-Order Focused Logic of Delimited Continuations

Toward a Solution

Higher-Order Focused Logic of Delimited Continuations

- Zeilberger, 2010: an inversion term is a **continuation** that consumes a focus term (also Curien & Munch-Maccagnoni, 2010; M-M & Scherer, 2013)

Toward a Solution

Higher-Order Focused Logic of Delimited Continuations

- Zeilberger, 2010: an inversion term is a **continuation** that consumes a focus term (also Curien & Munch-Maccagnoni, 2010; M-M & Scherer, 2013)

- Formally, assume that $[P]$ and $N > P$ are defined. Then, where $(\Rightarrow)$ is the meta arrow, let:

Toward a Solution

Higher-Order Focused Logic of Delimited Continuations

- Zeilberger, 2010: an inversion term is a **continuation** that consumes a focus term (also Curien & Munch-Maccagnoni, 2010; M-M & Scherer, 2013)

- Formally, assume that $[P]$ and $N > P$ are defined. Then, where $(\Rightarrow)$ is the meta arrow, let:

  - Left inversion: $P', \ldots, Q' \vdash P \triangleq [P'] \Rightarrow \ldots \Rightarrow [Q'] \Rightarrow [P]$

Toward a Solution

Higher-Order Focused Logic of Delimited Continuations

- Zeilberger, 2010: an inversion term is a **continuation** that consumes a focus term (also Curien & Munch-Maccagnoni, 2010; M-M & Scherer, 2013)
- Formally, assume that $[P]$ and $N > P$ are defined. Then, where $(\Rightarrow)$ is the meta arrow, let:
  - Left inversion: $P', \ldots, Q' \vdash P \triangleq [P'] \Rightarrow \ldots \Rightarrow [Q'] \Rightarrow [P]$
  - Right inversion: $\Gamma^+ \vdash N \triangleq$ for all $P : N > P \Rightarrow \Gamma^+ \vdash [P]$

Toward a Solution

Higher-Order Focused Logic of Delimited Continuations

- Zeilberger, 2010: an inversion term is a **continuation** that consumes a focus term (also Curien & Munch-Maccagnoni, 2010; M-M & Scherer, 2013)

- Formally, assume that $[P]$ and $N > P$ are defined. Then, where $(\Rightarrow)$ is the meta arrow, let:

  - Left inversion: $P', \ldots, Q' \vdash P \triangleq [P'] \Rightarrow \ldots \Rightarrow [Q'] \Rightarrow [P]$

  - Right inversion: $\Gamma^+ \vdash N \triangleq$ for all $P : N > P \Rightarrow \Gamma^+ \vdash [P]$

- Why bother?

Toward a Solution

Higher-Order Focused Logic of Delimited Continuations

- Zeilberger, 2010: an inversion term is a **continuation** that consumes a focus term (also Curien & Munch-Maccagnoni, 2010; M-M & Scherer, 2013)

- Formally, assume that $[P]$ and $N > P$ are defined. Then, where $(\Rightarrow)$ is the meta arrow, let:

  - Left inversion: $P', \ldots, Q' \vdash P \triangleq [P'] \Rightarrow \ldots \Rightarrow [Q'] \Rightarrow [P]$

  - Right inversion: $\Gamma^+ \vdash N \triangleq$ for all $P : N > P \Rightarrow \Gamma^+ \vdash [P]$

- Why bother?

  - Polarity arises from **value-continuation distinction**, not rule (non)backtracking

Toward a Solution

Higher-Order Focused Logic of Delimited Continuations

- Zeilberger, 2010: an inversion term is a **continuation** that consumes a focus term (also Curien & Munch-Maccagnoni, 2010; M-M & Scherer, 2013)

- Formally, assume that $[P]$ and $N > P$ are defined. Then, where $(\Rightarrow)$ is the meta arrow, let:

  - Left inversion: $P', \ldots, Q' \vdash P \triangleq [P'] \Rightarrow \ldots \Rightarrow [Q'] \Rightarrow [P]$

  - Right inversion: $\Gamma^+ \vdash N \triangleq$ for all $P : N > P \Rightarrow \Gamma^+ \vdash [P]$

- Why bother?

  - Polarity arises from **value-continuation distinction**, not rule (non)backtracking

  - However, **defunctionalization** recovers (first-order, CPS'd) focusing - see Zeilberger, 2011; M-M's thesis, ch. 3

The Solution

Type Structure of Extended Logic of DC $\simeq$ EEC

The Solution

Type Structure of Extended Logic of DC $\simeq$ EEC

- Zeilberger, 2010: right focus internalizing left focus $\Rightarrow$ positive type of stacks

$$\frac{N > P}{\left[P/N\right]}$$

The Solution

Type Structure of Extended Logic of DC $\simeq$ EEC

- Zeilberger, 2010: right focus internalizing left focus $\Rightarrow$ positive type of stacks
- **Our contribution:** linear functions, other EEC connectives

$$\frac{N > P}{\left[P/N\right]} \qquad \frac{\text{for all } P:\ O > P \Rightarrow N > P}{N \gg O} \qquad \frac{N \gg O}{[N \multimap O]}$$

Metatheory
& Commentary

Metatheory

& Commentary

- Term correspondence?

Metatheory

& Commentary

- Term correspondence?
  - *Weak* left focus is not primitive b/c negative codomain ⇒ right inversion would be circularly defined

Metatheory

& Commentary

- Term correspondence?
  - *Weak* left focus is not primitive b/c negative codomain $\Rightarrow$ right inversion would be circularly defined
  - So, linear functions are **stack transformers** (Barrett et al., 2023)

Metatheory

& Commentary

- Term correspondence?
  - *Weak* left focus is not primitive b/c negative codomain ⇒ right inversion would be circularly defined
  - So, linear functions are **stack transformers** (Barrett et al., 2023)
- Five cut, two identity rules: all admissible (easily)

Metatheory
& Commentary

- Term correspondence?
  - *Weak* left focus is not primitive b/c negative codomain ⇒ right inversion would be circularly defined
  - So, linear functions are **stack transformers** (Barrett et al., 2023)
- Five cut, two identity rules: all admissible (easily)
- Categorical semantics? More on that later…

Conclusion

Mostly Future Work

Conclusion

Mostly Future Work

- “Straightforward” to generalize to polymorphic, dependent types; corresponds to:

Conclusion

Mostly Future Work

- “Straightforward” to generalize to polymorphic, dependent types; corresponds to:
  - Polymorphic CBPV/EEC (Møgelberg & Simpson, 2009; Rioux & Zdancewic, 2020; Jiang et al. 2025)

Conclusion

Mostly Future Work

- "Straightforward" to generalize to polymorphic, dependent types; corresponds to:
  - Polymorphic CBPV/EEC (Møgelberg & Simpson, 2009; Rioux & Zdancewic, 2020; Jiang et al. 2025)
  - Focused linear dependent types, analogous to eMLTT (Ahman, 2017)

Conclusion

Mostly Future Work

- “Straightforward” to generalize to polymorphic, dependent types; corresponds to:
  - Polymorphic CBPV/EEC (Møgelberg & Simpson, 2009; Rioux & Zdancewic, 2020; Jiang et al. 2025)
  - Focused linear dependent types, analogous to eMLTT (Ahman, 2017)
- “Straightforward” to **defunctionalize** the calculus

Conclusion

Mostly Future Work

- "Straightforward" to generalize to polymorphic, dependent types; corresponds to:
  - Polymorphic CBPV/EEC (Møgelberg & Simpson, 2009; Rioux & Zdancewic, 2020; Jiang et al. 2025)
  - Focused linear dependent types, analogous to eMLTT (Ahman, 2017)
- "Straightforward" to **defunctionalize** the calculus
  - Would appear closer to EEC, L-calculi ⇒ categorical semantics by translation?

Conclusion

Mostly Future Work

- "Straightforward" to generalize to polymorphic, dependent types; corresponds to:
  - Polymorphic CBPV/EEC (Møgelberg & Simpson, 2009; Rioux & Zdancewic, 2020; Jiang et al. 2025)
  - Focused linear dependent types, analogous to eMLTT (Ahman, 2017)
- "Straightforward" to **defunctionalize** the calculus
  - Would appear closer to EEC, L-calculi ⇒ categorical semantics by translation?
- Working on relationship between stacks and **linear lenses**

Thanks!

ssomayya@alumni.cmu.edu