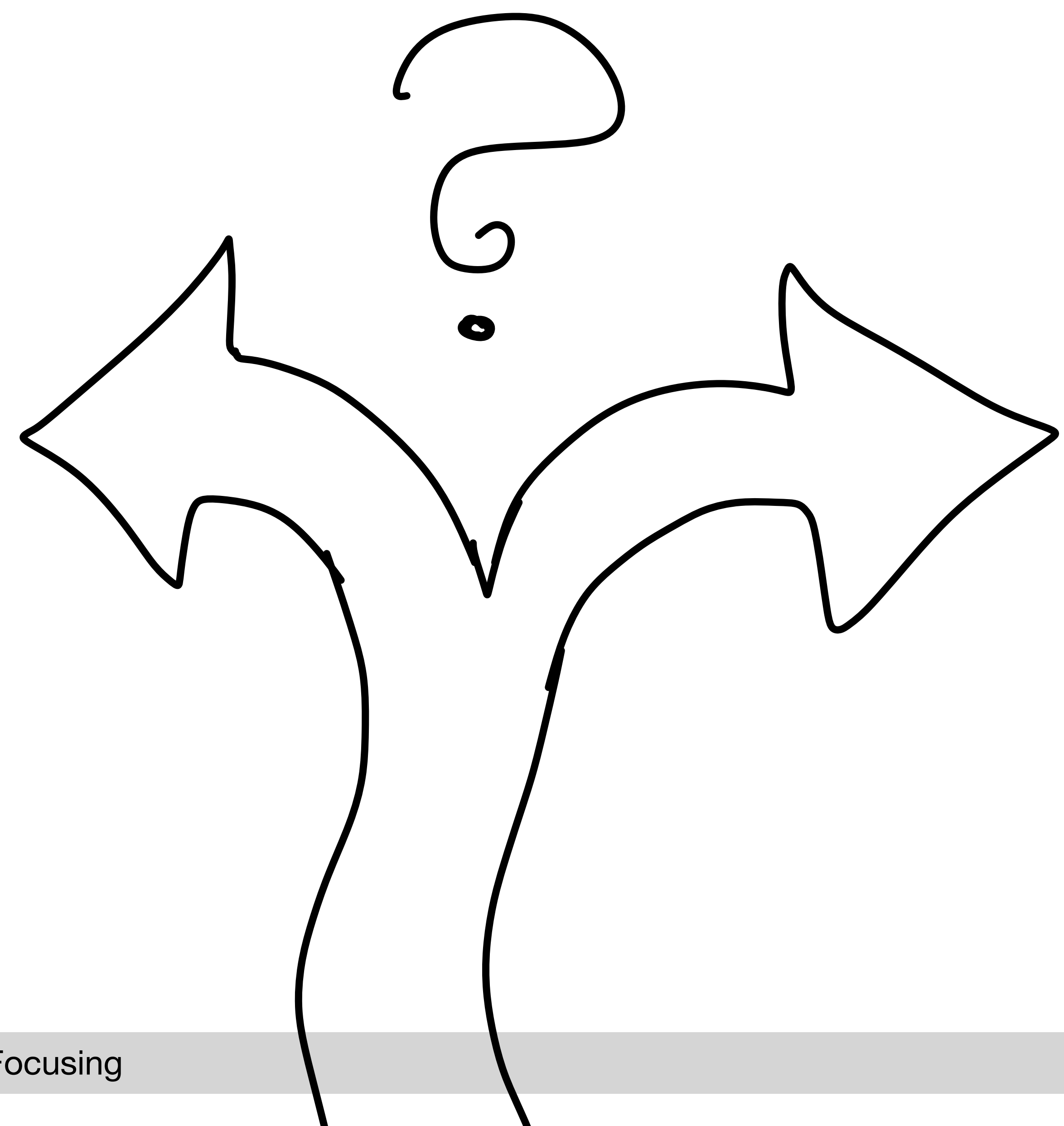
A guided tour of Polarity & Focusing

Chris Martens

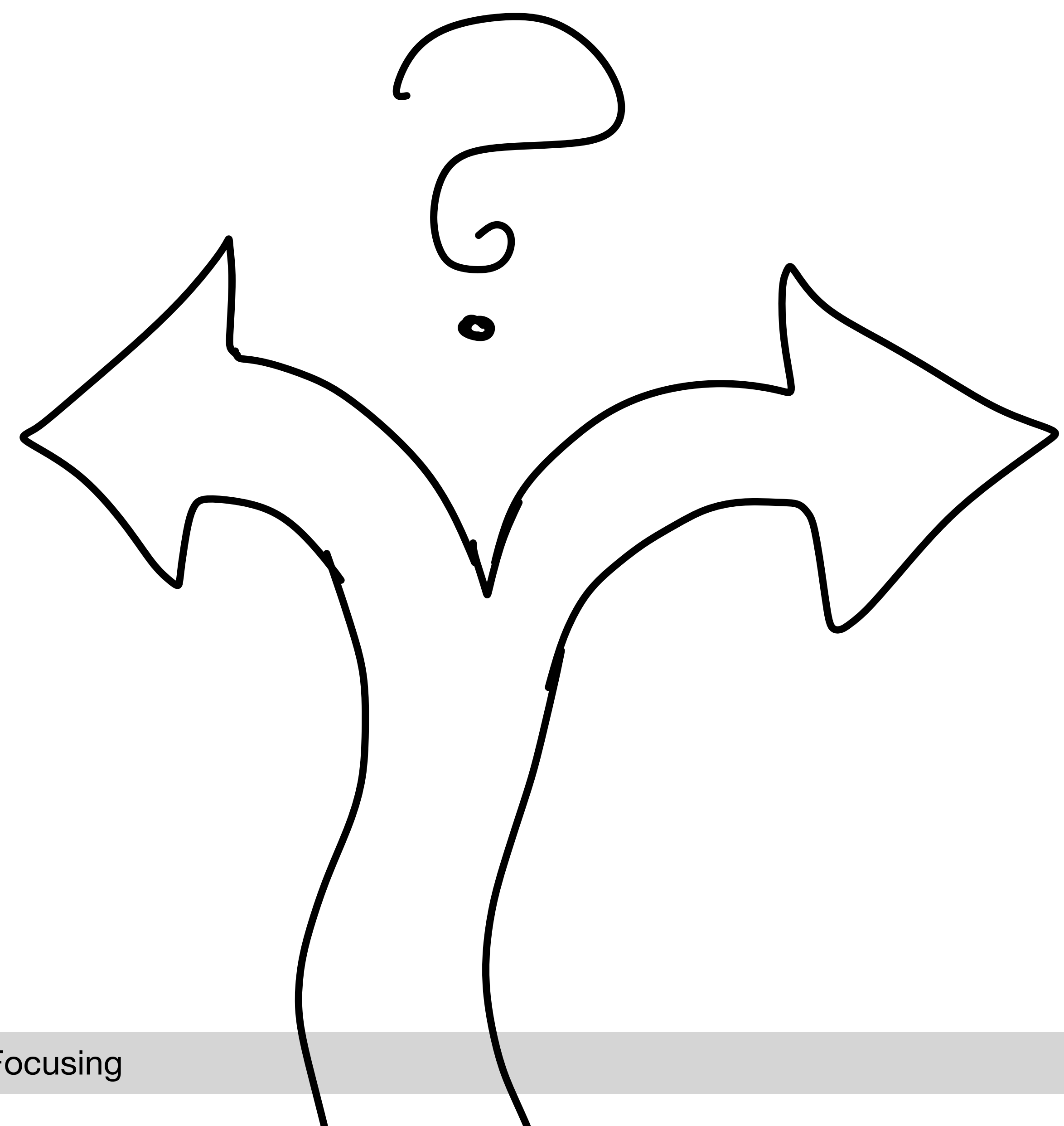
TYPES 2025

Life involves making choices.



Life involves making choices.

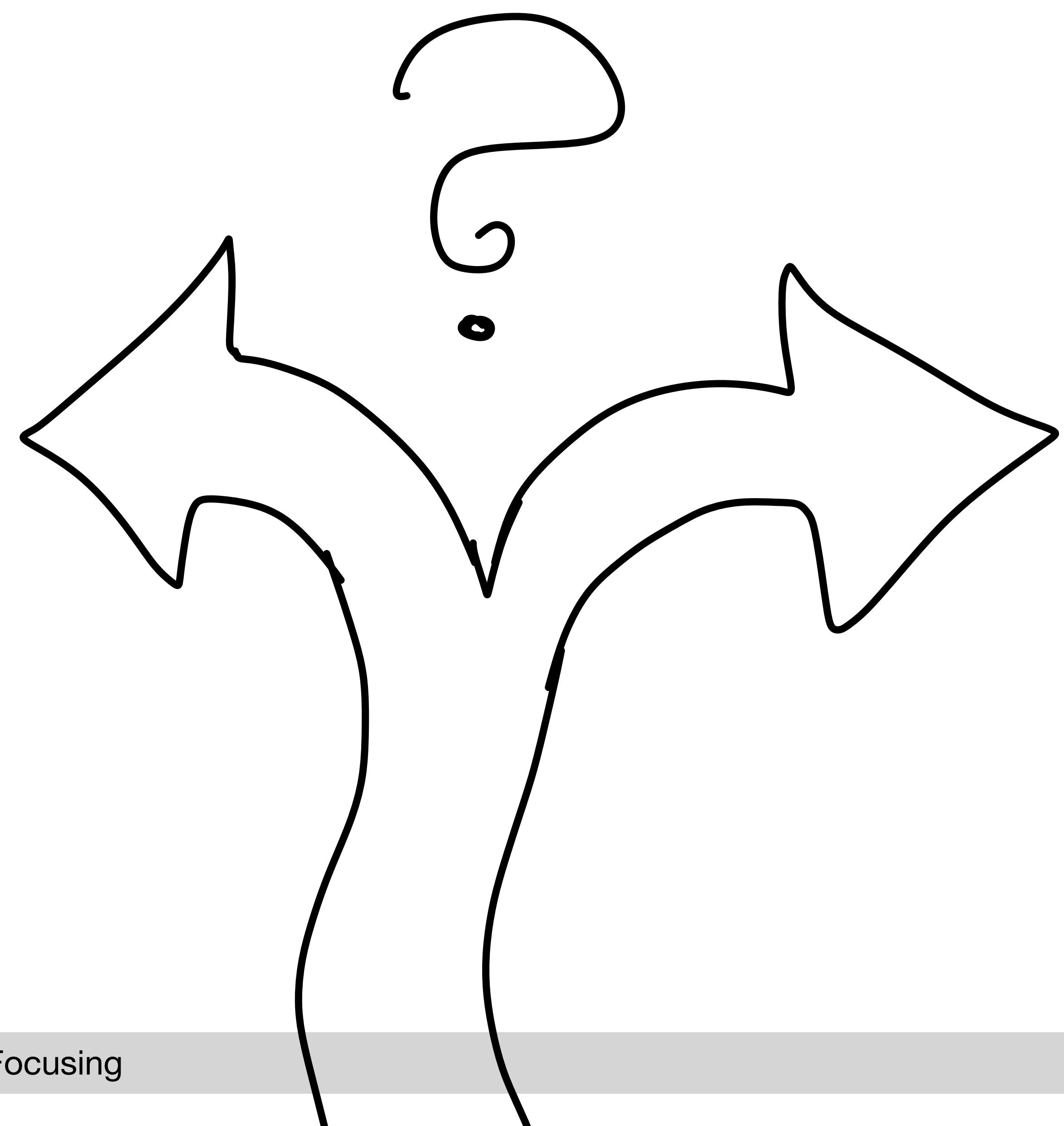
so does
proving theorems



Life involves making choices.

so does proving theorems

and writing programs.

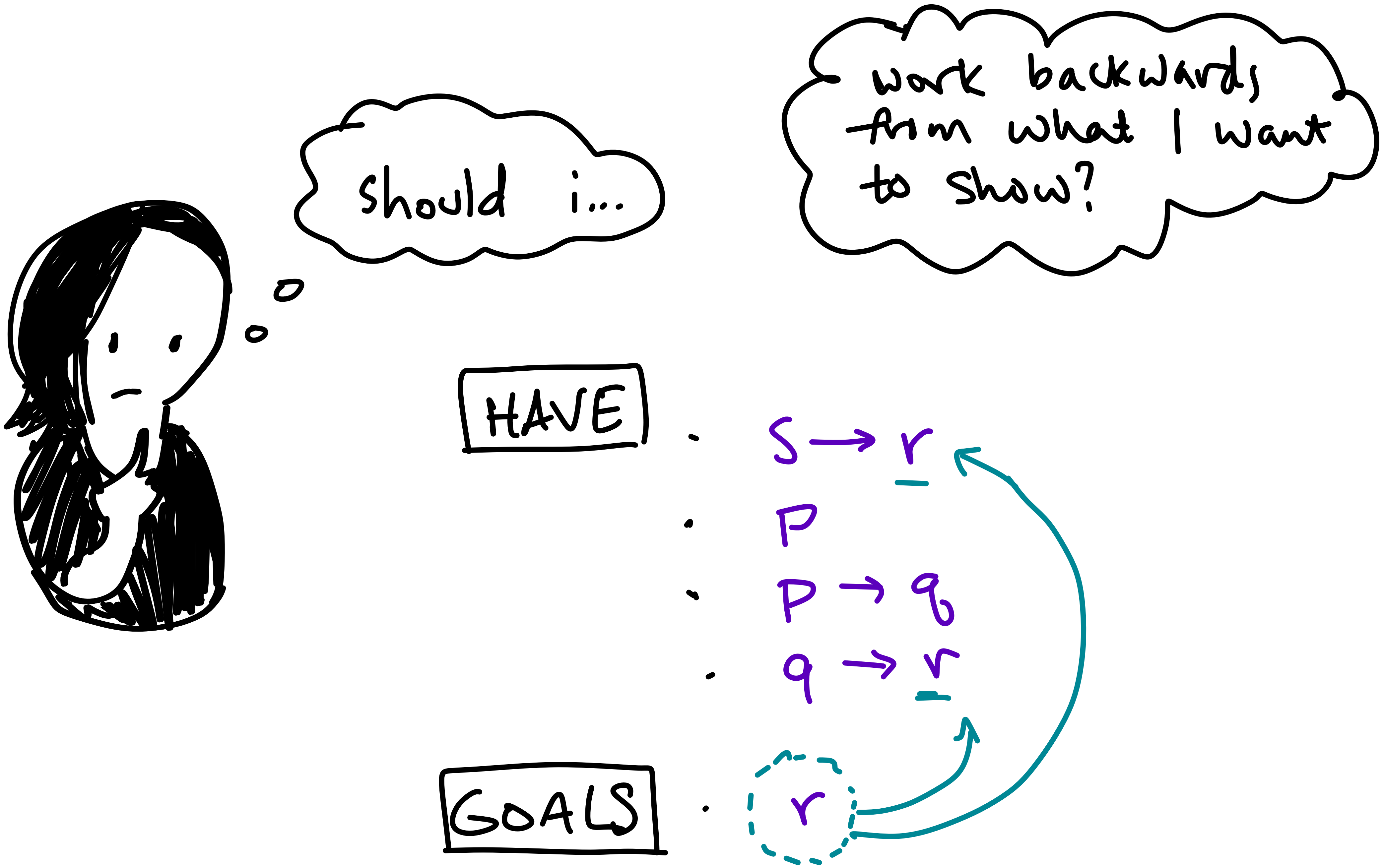


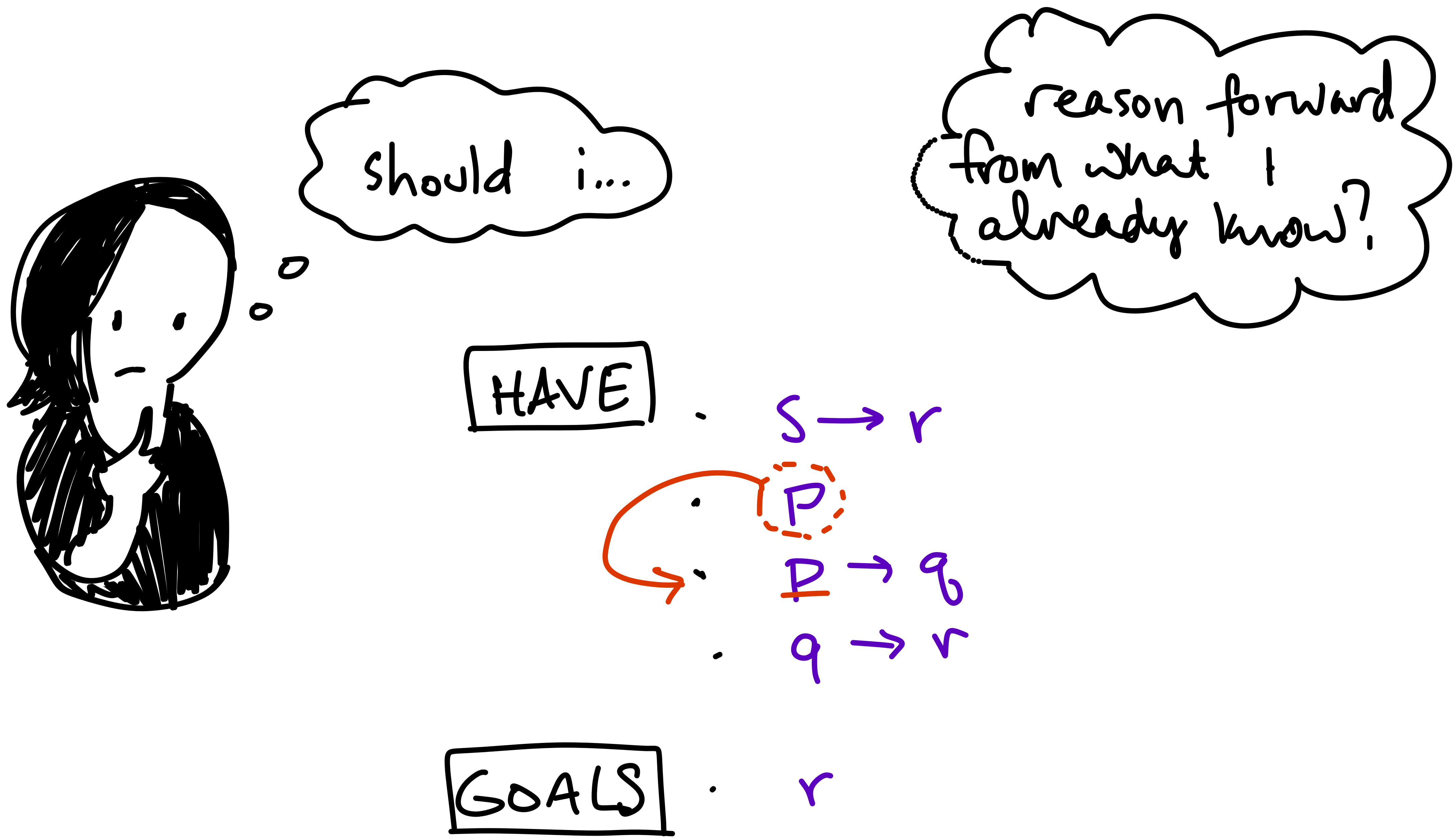
HAVE

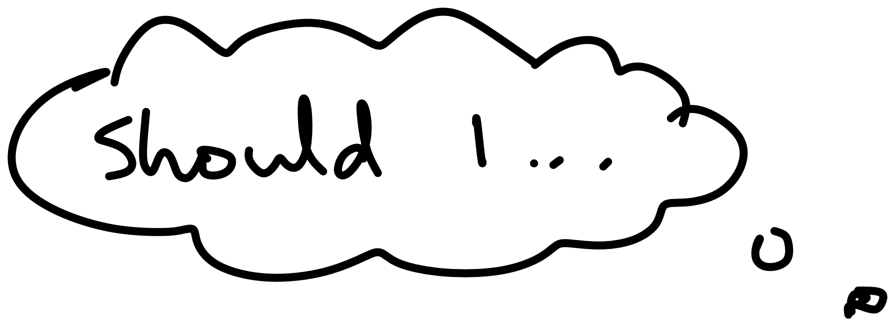
- $s \to r$
- p
- $p \to q$
- $q \to r$

GOALS

- r


should i...
work backwards from what I want to show?
HAVE
s → r
p
p → q
q → r
GOALS
r


should i...
reason forward from what I already know?
HAVE
. s → r
. p
. p → q
. q → r
GOALS
. r



$$\text{update} : \text{World} \times \text{Action} \to \text{World}$$

$$\mathrm{update} : \mathrm{World} \times \mathrm{Action} \to \mathrm{World}$$

$$\underline{\mathrm{type}}\ \mathrm{World} = \cancel{\mathrm{List}}\,(\mathrm{pos} \times \mathrm{entity})$$

Should I ...

use inductive data types?

or objects?

```
type World = Obj {
    update : Action → ()
    loadMap : map → ()
    getPos : entity → pos
    setPos : entity × pos → ()
}
```

Polarity & focusing

are proof-theoretic methods

that can help us

understand the consequences

of these choices.

As seen in:

<u>Values</u>

$()$ (v_1, v_2)

$in_1\ v$ $in_2\ v$

<u>Computations</u>

$\lambda x.e$ $f(e)$

$\pi_1\ e$ $\pi_2\ e$

$\underline{do}\ x \leftarrow e\ \underline{in}\ e'$

Values	Computations
- observable structure	- opaque
- use by pattern matching	- use by providing arguments

"positive"

Values

- observable structure
- use by pattern matching

"negative"

Computations

- opaque
- use by providing arguments

But: why are these polarities assigned to these types?

What does it have to do with making choices in proof search?

GOALS of THIS TALK

- Demystify polarity & focusing to the uninitiated;
- Recount some history and lineage, stopping to see the sights
- Grapple with some persistent myths & misconception;
- Motivate continued study.

Proof search

Sequent calculus

$$\underbrace{A_1, \ldots, A_n}_{\Gamma} \Longrightarrow A$$

Sequent calculus

$$\frac{\vdots \qquad\qquad \vdots}{\underbrace{A_1, \ldots, A_n}_{\Gamma} \Longrightarrow A}$$

Sequent calculus

$$\frac{\vdots \qquad\qquad \vdots}{\underbrace{A_1, \ldots, A_n}_{\Gamma} \Longrightarrow A}$$

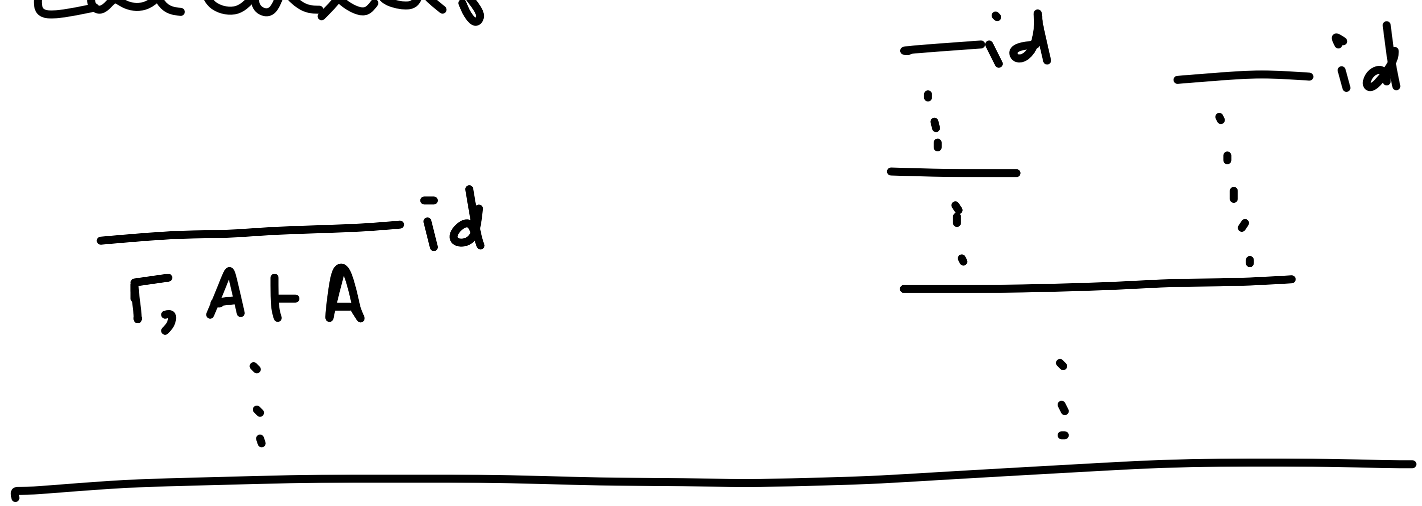
right rules

Sequent calculus

left rules

$$\dfrac{\vdots \qquad\qquad \vdots}{\underbrace{A_1, \ldots, A_n}_{\Gamma} \Longrightarrow A}$$

Sequent calculus



$$\underbrace{A_1, \ldots, A_n}_{\Gamma} \Longrightarrow A$$

Sequent calculus, propositional, intuitionistic.

$$A, B ::= p \mid \mathbb{1} \mid A \times B \mid A \to B$$

Sequent calculus, propositional, intuitionistic.

$$\begin{array}{lcccccccc} A, B & ::= & p & \mid & \mathbb{1} & \mid & A \times B & \mid & A \to B \\ & & & & \top & & \wedge & & \supset \end{array}$$

$$\frac{}{\Gamma, A \vdash A}\ \text{id}$$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \times B}\ \times R \qquad \frac{\Gamma, A, B \vdash C}{\Gamma, A \times B \vdash C}\ \times L$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}\ \to R \qquad \frac{\Gamma, A \to B \vdash A \quad \Gamma, A \to B, B \vdash C}{\Gamma, A \to B \vdash C}\ \to L$$

$$\frac{}{\Gamma \vdash 1}\ 1R \qquad \frac{\Gamma \vdash C}{\Gamma, 1 \vdash C}\ 1L$$

$$p \times q,\ q \times r \to s \vdash r \to s$$

$$\frac{}{\Gamma, A \vdash A}\,\mathrm{id}$$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \times B}\,\times R \qquad \frac{\Gamma, A, B \vdash C}{\Gamma, A \times B \vdash C}\,\times L$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}\,\to R \qquad \frac{\Gamma, A \to B \vdash A \quad \Gamma, A \to B, B \vdash C}{\Gamma, A \to B \vdash C}\,\to L$$

$$\frac{}{\Gamma \vdash \mathbb{1}}\,\mathbb{1}R \qquad \frac{\Gamma \vdash C}{\Gamma, \mathbb{1} \vdash C}\,\mathbb{1}L$$

$$p \times q,\; q \times r \to s \;\vdash\; r \to s$$

$$\frac{}{\Gamma, A \vdash A}\ \mathrm{id}$$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \times B}\ \times R \qquad \frac{\Gamma, A, B \vdash C}{\Gamma, A \times B \vdash C}\ \times L$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}\ \to R \qquad \frac{\Gamma, A \to B \vdash A \quad \Gamma, A \to B, B \vdash C}{\Gamma, A \to B \vdash C}\ \to L$$

$$\frac{}{\Gamma \vdash 1}\ 1R \qquad \frac{\Gamma \vdash C}{\Gamma, 1 \vdash C}\ 1L$$

$$p \times q,\ q \times r \to s \vdash r \to s$$

$$\frac{}{\Gamma, A \vdash A}\ \mathrm{id}$$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \times B}\ \times R \qquad \frac{\Gamma, A, B \vdash C}{\Gamma, A \times B \vdash C}\ \times L$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}\ \to R \qquad \frac{\Gamma, A \to B \vdash A \quad \Gamma, A \to B, B \vdash C}{\Gamma, A \to B \vdash C}\ \to L$$

$$\frac{}{\Gamma \vdash \mathbb{1}}\ \mathbb{1}R \qquad \frac{\Gamma \vdash C}{\Gamma, \mathbb{1} \vdash C}\ \mathbb{1}L$$

$$p \times q,\ q \times r \to s \ \vdash\ r \to s$$

$$\frac{}{\Gamma, A \vdash A}\,\mathrm{id}$$

$$\frac{\Gamma \vdash A \qquad \Gamma \vdash B}{\Gamma \vdash A \times B}\,{\times}R \qquad \frac{\Gamma, A, B \vdash C}{\Gamma, A \times B \vdash C}\,{\times}L$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}\,{\to}R \qquad \frac{\Gamma, A \to B \vdash A \qquad \Gamma, A \to B, B \vdash C}{\Gamma, A \to B \vdash C}\,{\to}L$$

$$\frac{}{\Gamma \vdash \mathbb{1}}\,\mathbb{1}R \qquad \frac{\Gamma \vdash C}{\Gamma, \mathbb{1} \vdash C}\,\mathbb{1}L$$

$$\frac{\Gamma \vdash q \times r \qquad \vdots}{\Gamma = p \times q,\ \underline{q \times r \to s} \vdash r \to s}\,{\to}L$$

$$\frac{}{\Gamma, A \vdash A}\ \text{id}$$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \times B}\ \times R \qquad \frac{\Gamma, A, B \vdash C}{\Gamma, A \times B \vdash C}\ \times L$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}\ {\to}R \qquad \frac{\Gamma, A \to B \vdash A \quad \Gamma, A \to B, B \vdash C}{\Gamma, A \to B \vdash C}\ {\to}L$$

$$\frac{}{\Gamma \vdash 1}\ 1R \qquad \frac{\Gamma \vdash C}{\Gamma, 1 \vdash C}\ 1L$$

$\checkmark$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma \vdash q}\ \text{id} \quad \dfrac{}{\Gamma \vdash r}\ \text{id}}{\Gamma \vdash q \times r}\ \times R \qquad \dfrac{}{\Gamma, s \vdash s}\ \text{id}}{\Gamma:\ p, q, q \times r \to s, r \vdash s}\ {\to}L}{p \times q, q \times r \to s, r \vdash s}\ \times L}{p \times q, q \times r \to s \vdash r \to s}\ {\to}R$$

$$\frac{}{\Gamma, A \vdash A}\,\text{id}$$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \times B}\,\times R \qquad \frac{\Gamma, A, B \vdash C}{\Gamma, A \times B \vdash C}\,\times L$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}\,\to R \qquad \frac{\Gamma, A \to B \vdash A \quad \Gamma, A \to B, B \vdash C}{\Gamma, A \to B \vdash C}\,\to L$$

$$\frac{}{\Gamma \vdash 1}\,1R \qquad \frac{\Gamma \vdash C}{\Gamma, 1 \vdash C}\,1L$$

$$\frac{\dfrac{\dfrac{\dfrac{\dfrac{}{p, q, \ldots \vdash q}\,\text{id}}{\Gamma \vdash q}\,\times L \quad \dfrac{}{\Gamma \vdash r}\,\text{id}}{\Gamma \vdash q \times r}\,\times R \qquad \dfrac{}{\Gamma, s \vdash s}\,\text{id}}{\Gamma = p \times q,\ q \times r \to s,\ r \vdash s}\,\to L}{p \times q,\ q \times r \to s \vdash r \to s}\,\to R$$

Context: logic programming — proof search as computation

```
q :- p.
p.
?q
```

$$\boxed{p, p \to q \vdash q}$$

"Logic programming with focusing proofs in linear logic." Jean-Marc Andreolli, J. Logic and Computation, 1992.

Context: logic programming —
proof search as computation

How to cut down
nondeterminism in proof search?

OBSERVATION (Andreoli '92):

Some rules are $\boxed{\text{invertible}}$:

conclusion implies premises

$$\frac{\Gamma, A, B \vdash C}{\Gamma, A \times B \vdash C}$$

OBSERVATION (Andreoli '92):

Some rules are $\boxed{\text{invertible}}$:

conclusion implies premises

$$\frac{\Gamma, A, B \vdash C}{\Gamma, A \times B \vdash C}$$

it should always be safe to apply these!

Non-invertible rules represent important choices —

$$\frac{\Gamma, A \to B \vdash A \qquad \Gamma, A \to B, B \vdash C}{\Gamma, A \to B \vdash C} \to L$$

— we may have to backtrack.

Two things to try:

1 Apply invertible rules eagerly.

Two things to try:

1. Apply invertible rules eagerly.

2. When we decide to try a non-invertible rule, <u>hard</u> <u>commit</u>.

That is, if we pick $\to L$ on this formula,

$$\frac{\Gamma, A \to B \vdash A \qquad \Gamma, A \to B, B \vdash C}{\Gamma, [A \to B] \vdash C} \to L$$

That is, if we pick $\to L$ on this formula,

$$\frac{\Gamma, A \to B \vdash \boxed{A} \qquad \Gamma, A \to B, \boxed{B} \vdash C}{\Gamma, \boxed{A \to B} \vdash C} \to L$$

stay with the subformulas as long as possible.

Focusing (proof search strategy):

1. Apply all invertible rules. (inversion phase)
2. Pick a formula to focus on. (focus phase
3. Continue applying non-invertible rules to its subformulas until none apply.)
4. Go to 1.

Focusing (proof search strategy):

1. Apply all invertible rules. (inversion phase)
2. Pick a formula to focus on. (focus phase
3. Continue applying non-invertible rules to its subformulae until none apply.)
4. Go to 1.

SURPRISING RESULT: this strategy is complete

(for linear, classical, intuitionistic ... logics)

Example.

$$p \times q,\ q \times r \to s \vdash r \to s$$

Example.

right inversion

$$\dfrac{p \times q,\ q \times r \to s,\ \underline{r} \vdash s}{p \times q,\ q \times r \to s \vdash \underline{r \to s}} \to R$$

Example.

left inversion

$$\dfrac{\dfrac{\underline{p}, \underline{q}, q \times r \to s, r \vdash s}{\underline{p \times q}, q \times r \to s, r \vdash s}\ \times L}{p \times q, q \times r \to s \vdash r \to s}\ \to R$$

Example.

Left focus

$$
\dfrac{\dfrac{\dfrac{\Gamma \vdash [q \times r] \qquad \Gamma, [s] \vdash s}{\Gamma = p, q, [q \times r \to s], r \vdash s} \to L}{p \times q, q \times r \to s, r \vdash s} \times L}{p \times q, q \times r \to s \vdash r \to s} \to R
$$

Example. Right focus

$$
\dfrac{
  \dfrac{
    \dfrac{
      \dfrac{\Gamma \vdash [q] \quad \Gamma \vdash [r]}{\Gamma \vdash [q \times r]}\times R
      \qquad
      \Gamma, [s] \vdash s
    }{p, q, q \times r \to s, r \vdash s}\to L
  }{p \times q, q \times r \to s, r \vdash s}\times L
}{p \times q, q \times r \to s \vdash r \to s}\to R
$$

Example.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma \vdash q}\,id \quad \dfrac{}{\Gamma \vdash r}\,id}{\Gamma \vdash q \times r}\,{\times}R \qquad \dfrac{}{\Gamma, s \vdash s}\,id}{p, q, q \times r \to s, r \vdash s}\,{\to}L}{p \times q, q \times r \to s, r \vdash s}\,{\times}L}{p \times q, q \times r \to s \vdash r \to s}\,{\to}R$$

Example.

$$\cdot \vdash (p \to q) \to p \times r \to q \times r$$

Right invert

$$\frac{p \to q,\ p \times r \ \vdash\ q \times r}{\cdot \vdash (p \to q) \to p \times r \to q \times r} \to R^2$$

Left invert

$$\dfrac{\dfrac{p \to q,\ \underline{p},\ \underline{r} \vdash q \times r}{p \to q,\ \underline{p \times r} \vdash q \times r}\ \times L}{\cdot \vdash (p \to q) \to p \times r \to q \times r}\ {\to}R^2$$

Left focus

$$
\dfrac{\dfrac{\dfrac{p \to q, p, r \vdash [p] \qquad p \to q, p, r, [q] \vdash q \times r}{[p \to q], p, r \vdash q \times r} \to L}{p \to q, p \times r \vdash q \times r} \times L}{\cdot \vdash (p \to q) \to p \times r \to q \times r} \to R^2
$$

succeed

$$\dfrac{\dfrac{\dfrac{\dfrac{}{p \to q, \underline{p}, r \vdash \underline{p}}\;id \qquad p \to q, p, r, \boxed{q} \vdash q \times r}{p \to q, p, r \vdash q \times r}\;{\to}L}{p \to q, p \times r \vdash q \times r}\;{\times}L}{\cdot \vdash (p \to q) \to p \times r \to q \times r}\;{\to}R^2$$

blur

$$\dfrac{\dfrac{\dfrac{\overline{p\to q,\ p,\ r \vdash p}\;\text{id} \qquad \dfrac{p\to q,\ p,\ r,\ q \vdash q\times r}{p\to q,\ p,\ r,\ [q] \vdash q\times r}\;\text{blurL}}{p\to q,\ p,\ r \vdash q\times r}\;{\to}L}{p\to q,\ p\times r \vdash q\times r}\;{\times}L}{\cdot \vdash (p\to q)\to p\times r \to q\times r}\;{\to}R^2$$

right focus

$$
\dfrac{
  \dfrac{}{p \to q, \underline{p}, r \vdash \underline{p}}\,id
  \qquad
  \dfrac{
    \dfrac{p \to q, p, r, q \vdash [q \times r]}{p \to q, p, r, q \vdash q \times r}\,focR
  }{p \to q, p, r, [q] \vdash q \times r}\,blurL
}{
  \dfrac{p \to q,\ p,\ r \ \vdash\ q \times r}{
    \dfrac{p \to q,\ p \times r \ \vdash\ q \times r}{\cdot \vdash (p \to q) \to p \times r \to q \times r}\,{\to}R^2
  }\,{\times}L
}\,{\to}L
$$

right focus

$$\dfrac{\Gamma \vdash [q] \qquad \Gamma \vdash [r]}{\Gamma = p \to q, p, r, q \vdash [q \times r]}\ \times R$$

$$\dfrac{\Gamma = p \to q, p, r, q \vdash [q \times r]}{p \to q, p, r, q \vdash q \times r}\ \mathrm{focR}$$

$$\dfrac{p \to q, p, r, q \vdash q \times r}{p \to q, p, r, [q] \vdash q \times r}\ \mathrm{blurL}$$

$$\dfrac{}{p \to q, \underline{p}, r \vdash \underline{p}}\ \mathrm{id}$$

$$\dfrac{p \to q, \underline{p}, r \vdash \underline{p} \qquad p \to q, p, r, [q] \vdash q \times r}{p \to q,\ p,\ r \vdash q \times r}\ \to L$$

$$\dfrac{p \to q,\ p,\ r \vdash q \times r}{p \to q,\ p \times r \vdash q \times r}\ \times L$$

$$\dfrac{p \to q,\ p \times r \vdash q \times r}{\cdot \vdash (p \to q) \to p \times r \to q \times r}\ \to R^2$$

Succeed

$$
\dfrac{
  \dfrac{}{p \to q, \underline{p}, r \vdash \underline{p}}\,\mathrm{id}
  \qquad
  \dfrac{
    \dfrac{
      \dfrac{
        \dfrac{}{\Gamma \vdash [q]}\,\mathrm{id}
        \qquad
        \dfrac{}{\Gamma \vdash [r]}\,\mathrm{id}
      }{\Gamma = p \to q, p, r, q \vdash [q \times r]}\,\times R
    }{p \to q, p, r, q \vdash q \times r}\,\mathrm{focR}
  }{p \to q, p, r, [q] \vdash q \times r}\,\mathrm{blurL}
}{
  \dfrac{p \to q, \; p, \; r \vdash q \times r}{
    \dfrac{p \to q, \; p \times r \vdash q \times r}{\cdot \vdash (p \to q) \to p \times r \to q \times r}\,\to R^2
  }\,\times L
}\,\to L
$$

A focused proof is a normal form

for sequent proofs.

A focused sequent calculus is one in which yields focused proofs by construction.

A focused sequent calculus is one in which yields focused proofs by construction.

how to get there?

Invertibility is a property of <u>rules</u>, not <u>connectives</u>.

	Invertible left rule?	Invertible right rule?
$\times$	✓	✓
1	✓	✓
$\to$		✓

In linear logic, observe...

	Invertible left rule?	Invertible right rule?
$\otimes$	✓	
1	✓	
$\&$		✓
$\multimap$		✓
$\top$		✓

In linear logic, observe...

		Invertible left rule?	Invertible right rule?
$\otimes$	POSITIVE	✓	
1		✓	
$\&$	NEGATIVE		✓
$\multimap$			✓
$\top$			✓

Distinguish + and − "and" (and unit)

$$A^+ ::= 1 \mid A \times B$$

$$A^- ::= \top \mid A \,\&\, B \mid A \to B$$

$$A ::= A^+ \mid A^- \mid p$$

Remove bipolar propositions:

$$A^+ ::= 1 \mid A^+ \times B^+ \mid p^+$$
$$A^- ::= \top \mid A^- \& B^- \mid A^+ \to B^- \mid p^-$$
$$A ::= A^+ \mid A^-$$

…and add <u>shifts</u> ($\downarrow, \uparrow$) to annotate changes in polarity.

$$A^+ ::= 1 \mid A^+ \times B^+ \mid p^+ \mid \boxed{\downarrow A^-}$$

$$A^- ::= \top \mid A^- \,\&\, B^- \mid A^+ \to B^- \mid \boxed{\uparrow A^+}$$

$$A ::= A^+ \mid A^-$$

Note that this means a given prop. in the unfocused system...

$$(p \to q) \to p \times r \to q \times r$$

Note that this means a given prop. in the unfocused system...

will first need to choose a polarity for each atom...

$$(p^+ \to q^-) \to p^+ \times r^+ \to q^- \times r^+$$

Note that this means a given prop. in the unfocused system...

will first need to choose a polarity for each atom...

...and then add shifts.

$$\downarrow(p^+ \to q^-) \to p^+ \times r^+ \to \uparrow(\downarrow q^- \times r^+)$$

Note that this means a given prop. in the unfocused system...

will first need to choose a polarity for each atom...

...and then add shifts.

polarization strategy

Note that this means a given prop. in the unfocused system...

will first need to choose a polarity for each atom...

...and then add shifts.

polarization strategy

("all +" common default)

A focused sequent calculus (judgments).

$\Gamma \vdash [A^+]$ right focus

$\Gamma \mid [A^-] \vdash \gamma$ left focus

$\Gamma ; \Omega \vdash A^-$ right inversion

$\Gamma ; \Omega, A^+ \vdash \gamma$ left inversion

$(\gamma ::= p^- \mid A^+)$

Moving between phases: blurring

$$\frac{\Gamma; \cdot \vdash A^-}{\Gamma \vdash [\downarrow A^-]}\ \downarrow R$$

right inversion

right focus

$$\frac{\Gamma; A^+ \vdash \gamma}{\Gamma \mid [\uparrow A^+] \vdash \gamma}\ \uparrow L$$

left inversion

left focus

Moving between phases: focusing

$$\frac{\Gamma \vdash [A^+]}{\Gamma ; \cdot \vdash A^+}\ \mathrm{foc}R$$

$$\frac{\Gamma, A^- \mid [A^-] \vdash \gamma}{\Gamma, A^- ; \cdot \vdash \gamma}\ \mathrm{foc}L$$

$\Gamma \vdash [A^{+}]$	right focus	$\times R,\ \mathbb{1}R,\ +R$
$\Gamma \mid [A^{-}] \vdash \gamma$	left focus	$\to L,\ \& L$
$\Gamma ; \Omega \vdash A^{-}$	right inversion	$\to R,\ \& R,\ \top R$
$\Gamma ; \Omega, A^{+} \vdash \gamma$	left inversion	$\times L,\ \mathbb{1}L,\ +L,\ 0L$

$$
\dfrac{
  \dfrac{
    \dfrac{
      \dfrac{}{\Gamma \vdash [q]}\,id_R
      \qquad
      \dfrac{}{\Gamma \vdash [r]}\,id_R
    }{\Gamma \vdash [q \times r]}
    \qquad
    \dfrac{}{\Gamma \mid [s] \vdash s}\,id_L
  }{\Gamma \mid [q \times r \to s] \vdash s}\,\to L
}{\Gamma = p, q, q \times r \to s, r \,;\, \cdot \vdash s}\,focL
$$

focus

stable sequent

$$
\dfrac{
  \dfrac{
    \dfrac{p, q, q \times r \to s, r \,;\, \cdot \vdash s}{\cdot \,;\, p, q, q \times r \to s, r \vdash s}\,\downarrow L*
  }{\cdot \,;\, p \times q, q \times r \to s, r \vdash s}\,\times L
}{\cdot \,;\, p \times q, q \times r \to s \vdash r \to s}\,\to R
$$

inversion


so,
when do we get to write a program?

Programming

Computational Interpretations

1. LOGIC PROGRAMMING

   proof search
   as
   computation

2. λ-calculi

   proof <u>reduction</u>
   as
   computation

Logic programming: proof search as computation

- <u>forward chaining</u>: start from facts & apply rules eagerly.

```
path X Y :- edge X Y.
path X Y :- edge X Z, path Z Y.
```

- <u>backward chaining</u>: start from query & apply rules to find the answer.

```
-?- path a b
```

Logic programming: proof search as computation

- **forward chaining**: start from facts & apply rules eagerly.

"datalog"

```
path X Y :- edge X Y.
path X Y :- edge X Z, path Z Y.
```

- **backward chaining**: start from query & apply rules to find the answer.

"prolog"

```
-?- path a b
```

Logic programming: proof search as computation

- <u>forward chaining</u>:
  polarize all atoms +

- <u>backward chaining</u>:
  polarize all atoms −

2. λ calculi, AKA computation as proof reduction, AKA proofs as programs.

2. λ calculi, AKA computation as proof reduction, AKA proofs as programs.

Warmup example: $\&$ vs. $\times$

term $[e_1, e_2] : A \& B^-$

value $(V_1, V_2) : A^+ \times B^+$

use: $e \cdot \pi_1$

$e \cdot \pi_2$

use: <u>match</u> V <u>with</u>

$(x, y) \Rightarrow \ldots$

2. λ calculi, AKA computation as proof reduction, AKA proofs as programs.

Warmup example: $\&$ vs. $\times$

term $[e_1, e_2] : A^- \& B^-$

use: $e \cdot \pi_1$
$e \cdot \pi_2$

"lazy pair"

value $(v_1, v_2) : A^+ \times B^+$

use: <u>match</u> v <u>with</u> $(x, y) \Rightarrow \ldots$

"eager pair"

2. λ calculi, AKA computation as proof reduction, AKA proofs as programs.

$\Gamma \vdash [A^+]$	$\times R$, $\mathbb{1}R$, $+R$
$\Gamma \mid [A^-] \vdash \gamma$	$\to L$, $\& L$
$\Gamma ; \Omega \vdash A^-$	$\to R$, $\& R$, $\top R$
$\Gamma ; \Omega, A^+ \vdash \gamma$	$\times L$, $\mathbb{1}L$, $+L$, $\mathbb{0}L$

2. λ calculi, AKA computation as proof reduction, AKA proofs as programs.

$\Gamma \vdash V : [A^+]$	values V	$\times R$, $1R$, $+R$
$\Gamma \mid [A^-] \vdash S : \gamma$	spines S	$\to L$, $\& L$
$\Gamma ; \Omega \vdash e : A^-$	terms e	$\to R$, $\& R$, $\top R$
$\Gamma ; \Omega, A^+ \vdash p : \gamma$	patterns p	$\times L$, $1L$, $+L$, $0L$

Proof term for focused proof of

$\downarrow(p \to \uparrow q) \to p \times r \to \uparrow(q \times r)$:

$\lambda\{f : p \to \uparrow q\} \Rightarrow \lambda(x : p, y : r) \Rightarrow$

$\underline{\textbf{match}}\ (f \bullet x)\ \underline{\textbf{with}}$

$z : q \Rightarrow \underline{\textbf{return}}\ (z, y)$

Proof term for focused proof of

$\downarrow(p \to \uparrow q) \to p \times r \to \uparrow(q \times r)$:

$\lambda\{f : p \to \uparrow q\} \Rightarrow \lambda(x : p, y : r) \Rightarrow$

$\underline{\text{match}}\ (f \bullet x)\ \underline{\text{with}}$

$z : q \Rightarrow \underline{\text{return}}\ \langle z, y \rangle$

λs bind patterns

Proof term for focused proof of

$\downarrow(p \to \uparrow q) \to p \times r \to \uparrow(q \times r)$:

$\lambda\{f : p \to \uparrow q\} \Rightarrow \lambda(x : p, y : r) \Rightarrow$

$\underline{\text{match}}\ (f \bullet x)\ \underline{\text{with}}$

$z : q \Rightarrow \underline{\text{return}}\ \langle z, y\rangle$

λs bind patterns, as do matches

Proof term for focused proof of

$\downarrow(p \to \uparrow q) \to p \times r \to \uparrow(q \times r)$:

$\lambda\{f : p \to \uparrow q\} \Rightarrow \lambda(x : p,\ y : r) \Rightarrow$

$\underline{\text{match}}\ (f \bullet x)\ \underline{\text{with}}$

$z : q \Rightarrow \underline{\text{return}}\ \langle z, y\rangle$

functions must be fully applied
(η-expanded)

Proof term for focused proof of

$\downarrow(p \to \uparrow q) \to p \times r \to \uparrow(q \times r)$:

$\lambda\{f : p \to \uparrow q\} \Rightarrow \lambda(x : p,\ y : r) \Rightarrow$

<u>match</u> $(f \bullet x)$ <u>with</u>

$z : q \Rightarrow$ <u>return</u> (z, y)

analogous to monadic bind/sequencing

For more details, see:

- Dunfield & Krishnaswami "Bidir. Typing" 2021
  (closest to what I show here)
- Krishnaswami 2009. Focusing on Pattern Matching
  (1^{st} published account)
- Zeilberger diss. 2009 "The logical basis of ..."
- Simmons 2014 – "Structural Focalization"
  complete account of proof terms for
  fully focused system.

Much of this work in the context of dependent types for logical frameworks.

$$A\ M \stackrel{?}{\equiv} A\ M'$$

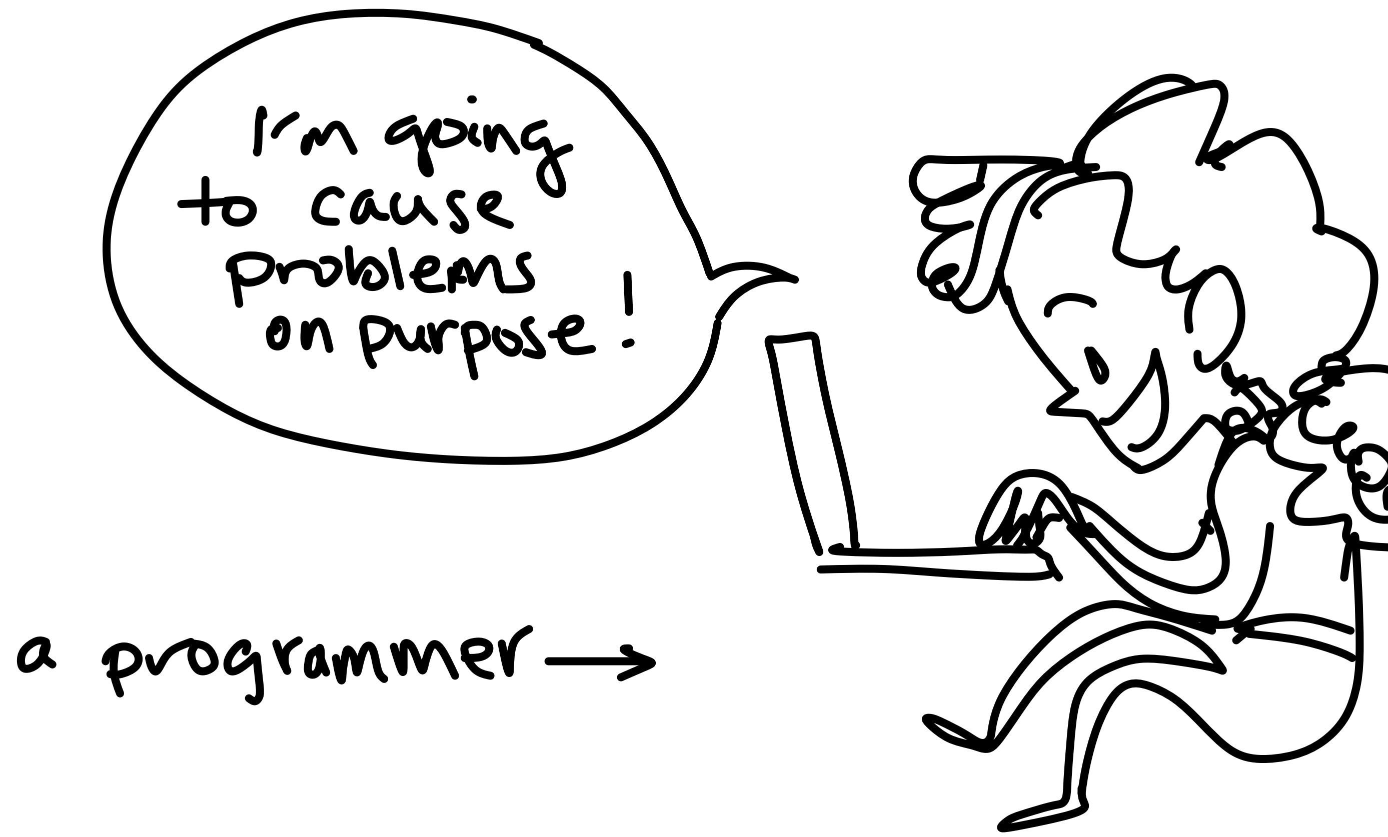
$$\Leftrightarrow \mathrm{nf}(M) = \mathrm{nf}(M')$$

"spine form" easier to check for equality:

$$h \cdot (t_1 ; t_2 ; t_3) \stackrel{?}{=} h' \cdot (t_1' ; t_2' ; t_3')$$

Normal forms

- focused proofs are "normal"
- programming adds redexes on purpose
- how to resolve?



Normal forms

- focused proofs are "normal"
- programming adds redexes on purpose
- how to resolve?

in SC, computation can be introduced with <u>cut</u>...

$$\frac{\Gamma \vdash A \qquad \Gamma', A \vdash B}{\Gamma', \Gamma \vdash B}$$

Normal forms

- focused proofs are "normal"
- programming adds redexes on purpose
- how to resolve?

in SC, computation can be introduced with <u>cut</u>...

$$\frac{\Gamma \vdash M : A \qquad \Gamma', x{:}A \vdash N : B}{\Gamma', \Gamma \vdash \mathsf{let}\ x{:}M\ \mathsf{in}\ N : B}$$

Normal forms

- focused proofs are "normal"
- programming adds redexes on purpose
- how to resolve?

The proof of <u>cut admissibility</u> (standard soundness criterion) yields a <u>cut elimination</u> procedure — i.e. program execution.

$$\dfrac{\dfrac{v_1 : A \quad v_2 : B}{(v_1, v_2) : A \times B} \qquad x{:}A, y{:}B \vdash e : C}{\underline{\text{match}}\ (v_1, v_2)\ \underline{\text{with}}\ (x,y) \to e \ : C} \longmapsto [v_1/x][v_2/y]e$$

Normal forms

- focused proofs are "normal"
- programming adds redexes on purpose
- how to resolve?

The proof of cut admissibility (standard soundness criterion) yields a cut elimination procedure — i.e. program execution.

- In LF, this corresponds to hereditary substitution.

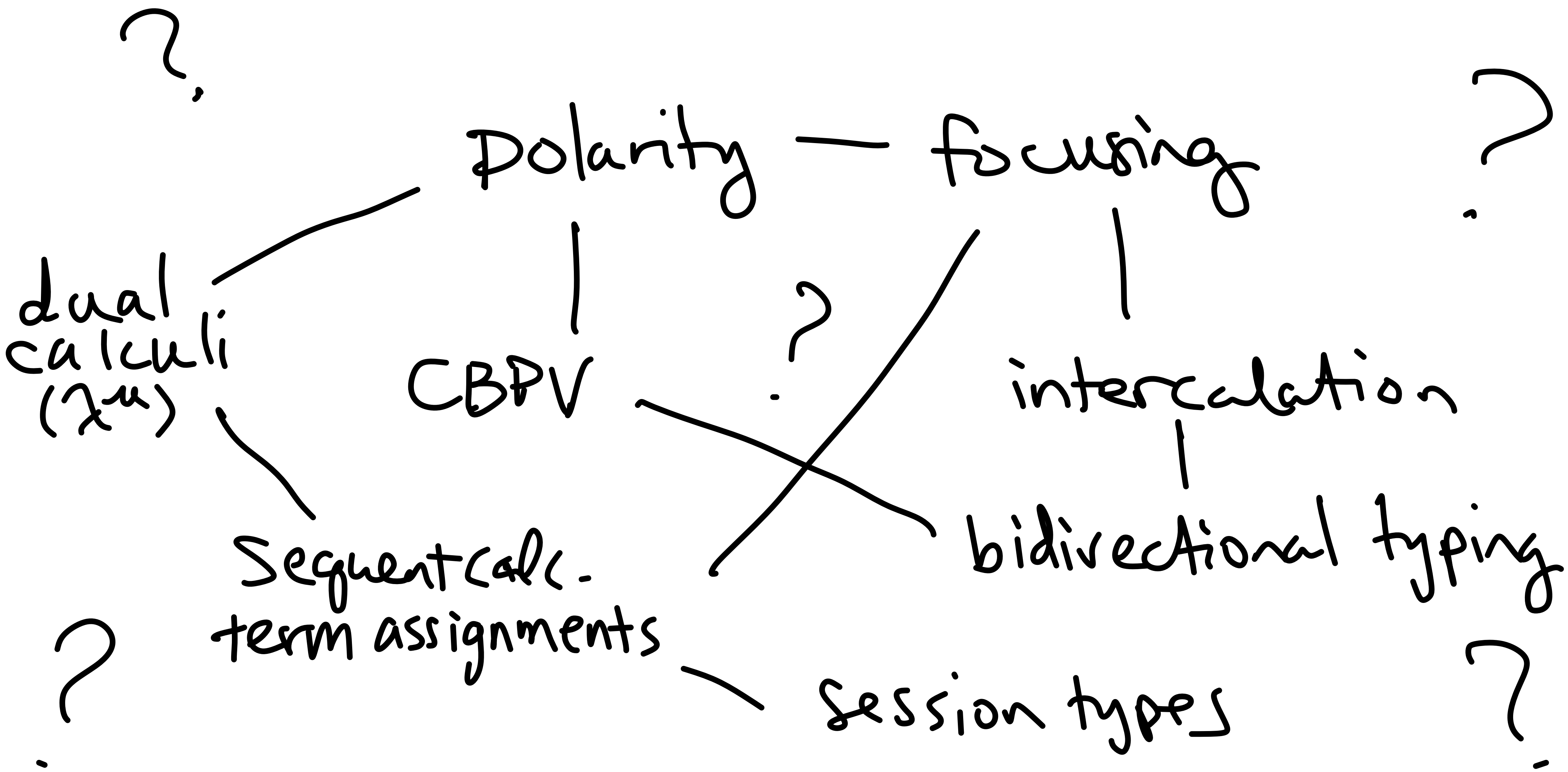
the strategy we use to make choices

affects the path traced

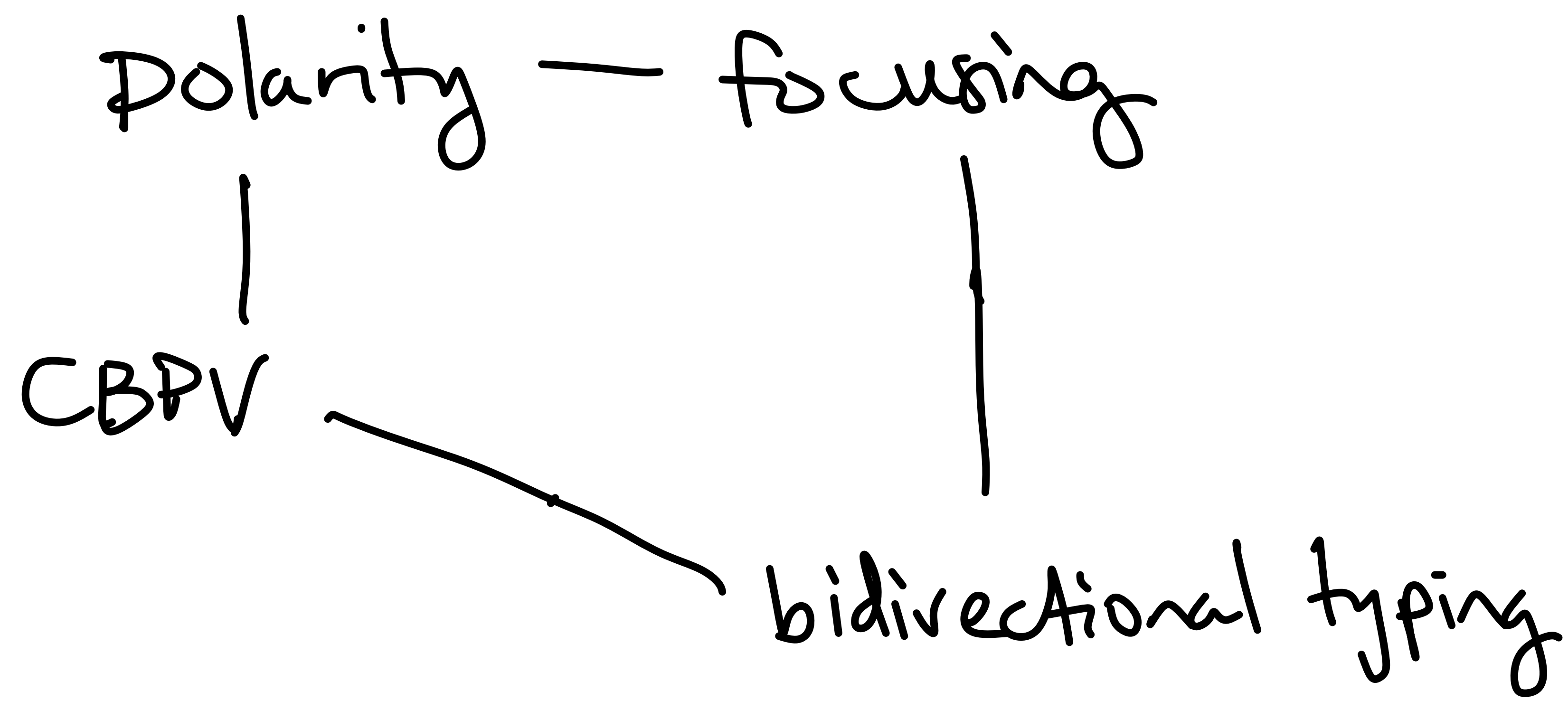
by our ~~footsteps~~.

Some points of frequent confusion and conflation

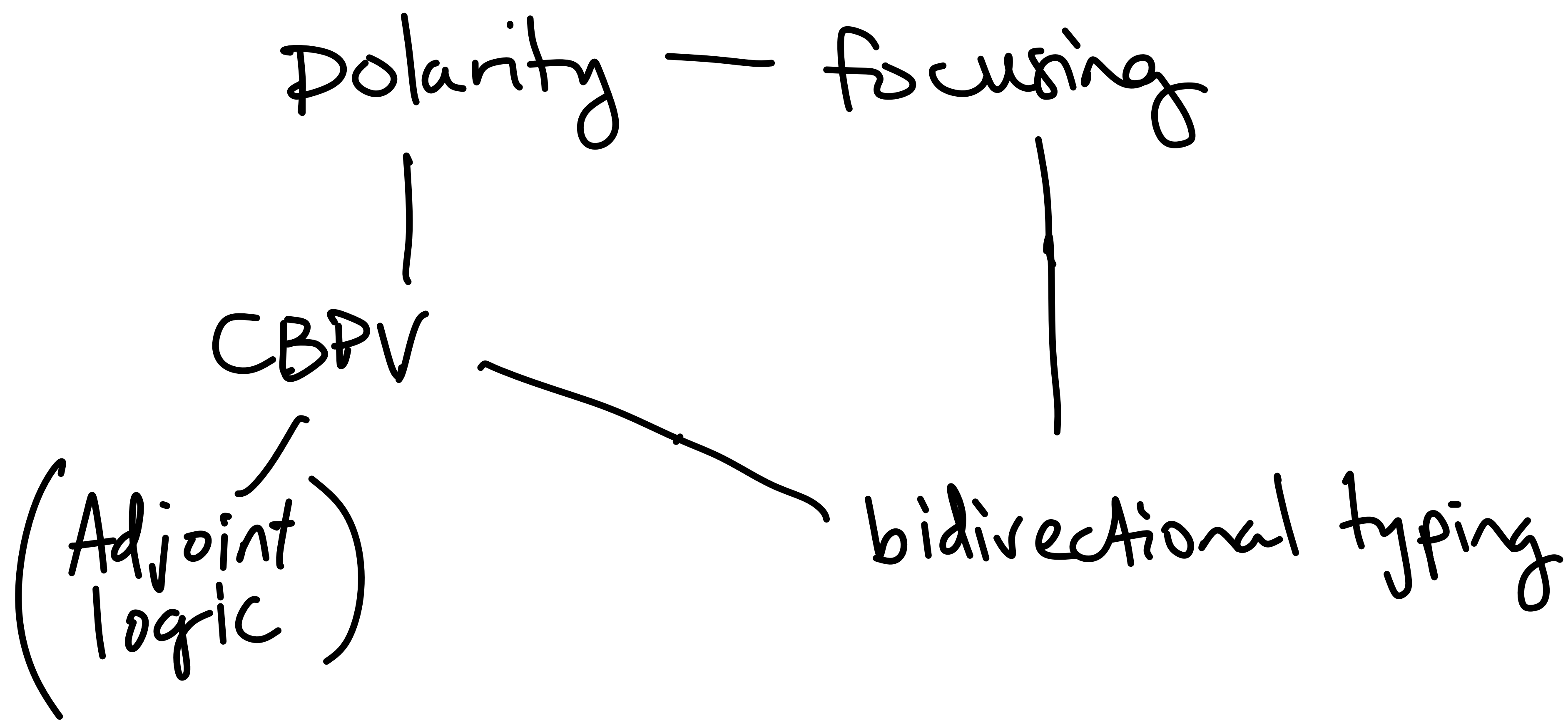
what are the relationships between ...



What are the relationships between ...



What are the relationships between ...



1. Bidirectional typing isn't the computational content of polarized logic.

- “Bidirectional Typing.” Jana Dunfield and Neel Krishnaswami. ACM Computing Surveys, 2021.

8 FOCUSING, POLARIZED TYPE THEORY, AND BIDIRECTIONAL TYPE SYSTEMS

A widespread folklore belief among researchers is that bidirectional typing arises from *polarized* formulations of logic. This belief is natural, helpful, and (surprisingly) wrong.

Bidirectional typing

$\Gamma \vdash e \Leftarrow A$

"e checks at A"

$\Gamma \vdash e \Rightarrow A$

"e synthesizes A"

Bidirectional typing

$\Gamma \vdash e \Leftarrow A$ "e checks at A"

$\Gamma \vdash e \Rightarrow A$ "e synthesizes A"

$$\frac{(x:A) \in \Gamma}{\Gamma \vdash x \Rightarrow A}\ \text{var}\Rightarrow$$

$$\frac{\Gamma \vdash e \Rightarrow A}{\Gamma \vdash e \Leftarrow A}\ \Rightarrow\Leftarrow$$

$$\frac{\Gamma, x:A \vdash e \Leftarrow B}{\Gamma \vdash \lambda x.e \Leftarrow A \to B}\ \to I\Leftarrow$$

$$\frac{\Gamma \vdash e_1 \Rightarrow A \to B \qquad \Gamma \vdash e_2 \Leftarrow A}{\Gamma \vdash e_1\, e_2 \Rightarrow B}\ \to E\Rightarrow$$

Bidirectional typing

$\Gamma \vdash e \Leftarrow A$ "e checks at A"

$\Gamma \vdash e \Rightarrow A$ "e synthesizes A"

$$\frac{(x : A) \in \Gamma}{\Gamma \vdash x \Rightarrow A}\ \mathrm{var}\Rightarrow$$

$$\frac{\Gamma \vdash e \Rightarrow A}{\Gamma \vdash e \Leftarrow A}\ \Rightarrow\Leftarrow$$

$$\frac{\Gamma, x : A \vdash e \Leftarrow B}{\Gamma \vdash \lambda x.e \Leftarrow A \to B}\ \to I\Leftarrow$$

$$\frac{\Gamma \vdash e_1 \Rightarrow A \to B \qquad \Gamma \vdash e_2 \Leftarrow A}{\Gamma \vdash e_1\, e_2 \Rightarrow B}\ \to E\Rightarrow$$

precisely characterizes β-η normal λ terms!

Abstract effects: sum types

$f : a \to b$

$x : a + a$

$\vdash b$

$$f\ (\text{match } x \;|\; x_1 \to x_1 \;|\; x_2 \to x_2)$$

vs.

$$\text{match } x \;|\; x_1 \to f\ x_1 \;|\; x_2 \to f\ x_2$$

- "Bidirectional Typing." Jana Dunfield and Neel Krishnaswami. ACM Computing Surveys, 2021.

At this point, we can make the following pair of observations:

(1) The simple bidirectional system for the STLC with products has the property that two terms are $\beta\eta$-equal if and only if they are the same: it fully characterizes $\beta\eta$-equality.
(2) Adding sum types to the bidirectional system breaks this property: two terms equivalent up to (some) commuting conversions may both be typable.

To restore this property, two approaches come to mind. The first approach is to find even more restrictive notions of normal form, which prohibit the commuting conversions. We will not pursue this direction in this article, but see Scherer [2017] and Ilik [2017] for examples of this approach.

The second approach is to find type theories in which the commuting conversions *no longer preserve equality*. By adding (abstract) effects to the language, terms that used to be equivalent can now be distinguished, ensuring that term equality once again coincides with semantic equality. This is the key idea embodied in what is variously called *polarized type theory*, *focalization*, or *call-by-push-value* [Levy 2001].

- “Bidirectional Typing.” Jana Dunfield and Neel Krishnaswami. ACM Computing Surveys, 2021.

At this point, we can make the following pair of observations:

(1) The simple bidirectional system for the STLC with products has the property that two terms are $\beta\eta$-equal if and only if they are the same: it fully characterizes $\beta\eta$-equality.
(2) Adding sum types to the bidirectional system breaks this property: two terms equivalent up to (some) commuting conversions may both be typable.

To restore this property, two approaches come to mind. The first approach is to find even more restrictive notions of normal form, which prohibit the commuting conversions. We will not pursue this direction in this article, but see Scherer [2017] and Ilik [2017] for examples of this approach.

The second approach is to find type theories in which the commuting conversions *no longer preserve equality*. By adding (abstract) effects to the language, terms that used to be equivalent can now be distinguished, ensuring that term equality once again coincides with semantic equality. This is the key idea embodied in what is variously called *polarized type theory*, *focalization*, or *call-by-push-value* [Levy 2001].

2. Polarity and focusing aren't the same thing.

computation/value type separation
with shifts

2. Polarity and focusing <u>aren't the
same thing</u>.

The ability to control the reduction strategy of a term through dedicated operators, reflected at the level of types by the presence of explicit *polarity shifts* — the key ingredient in the focalisation result — is a striking example of *double discovery*. Indeed, the focusing technique was a rather syntactic artifact of linear logic that rose to the status of *“éminence grise”* in proof theory, while the call-by-push-value language stems from the thorough analysis of the semantics of the two major functional paradigms. Their convergence is a sign of their significance.

- “Computation in focused intuitionistic logic.” Taus Brock-Nannestad, Nicolas Guenot, Daniel Gustafsson. PPDP 2015.

Call by Push Value (CBPV)

value types $A^+ ::= 1 \mid A^+ \times B^+ \mid U\,A^-$

Computation Types $A^- ::= A^+ \to B \mid F\,A^+$

$\Gamma ::= \cdot \mid \Gamma, x{:}A^+$

$\boxed{\Gamma \vdash v : A^+}$ $\boxed{\Gamma \vdash e : A^-}$

Call by Push Value (CBPV)

value types $A^+ ::= 1 \mid A^+ \times B^+ \mid U\,A^-$ positive

Computation Types $A^- ::= A^+ \to B \mid F\,A^+$ negative

$\Gamma ::= \cdot \mid \Gamma, x{:}A^+$

$\boxed{\Gamma \vdash v : A^+}$ $\boxed{\Gamma \vdash e : A^-}$

- “Computation in focused intuitionistic logic.” Taus Brock-Nannestad, Nicolas Guenot, Daniel Gustafsson. PPDP 2015.

5. Call-by-push-value and *LJF*

As we have seen in the previous sections, the ***LJF*** system offers a versatile framework for typing λ-terms extended by advanced constructs, providing at least a partial control over the reduction strategies. Of course, the introduction of shifts at the level of types and the encodings given for CBN and CBV are reminiscent of the *call-by-push-value* language [25] in which the markers U and F establish the distinction between *value types* and *computation types*. It appears clear that there is a connection here, but this raises the question: *are **LJF** and CBPV describing precisely the same language?* This section will show that they are almost the same, but not exactly.

From the perspective of our comparison, we see that CBPV and the $\lambda\kappa$-calculus, typed by ***LJF***, are not exactly the same: any term typeable in $\lambda\kappa$ must be η-long — up to atoms but also shifts, as for example $\downarrow N$ can be the type of some variable x — while in ***NJPV*** this restriction is not enforced. Of course, one could use the η-expansion result of CBPV, valid at non-atomic types, but it would

- “The bijection is between well-typed [focused λ terms] and CBPV terms where all subterms of function type are η-long.”
  (Brock-Nannestad et al. 2015)

2. Polarity & focusing <u>aren't the same thing</u>.

- CBPV: polarized but not focused.
- Krishnaswami 2009: focused, but not (explicitly) polarized (no $\uparrow\downarrow$).

Focusing gives you normal forms (β-short, η long)

Polarity gives you "abstract effects"

(wait a minute...)

We have adapted the CBPV syntax to fit our general framework, but one can easily see that U is $\downarrow$ and F is $\uparrow$, making $\lfloor\cdot\rfloor$ and $\lceil\cdot\rceil$ stand for *thunk* and *return* respectively, while F denotes the *forcing* of a value and the CBPV application $p`t$ is translated into $t\ p$. Formally, we use the following grammar:

$$\begin{array}{rcl} t, u & ::= & \lambda x.t \mid t\,p \mid \mathsf{F}\,p \mid \lceil p \rceil \mid u \;\mathsf{to}\; x.t \\ p, q & ::= & x \mid \lfloor t \rfloor \end{array}$$

and the terms $t\ p$ and $\mathsf{F}\,p$ are called *synthesised* terms, while the others are *checked* terms, in reference to the form of the typing rules.

$$\textit{\textbf{ax}}\ \frac{}{\Gamma, x : P \vDash x \Rightarrow P} \qquad \textit{\textbf{ni}}\ \frac{\Gamma \vDash p \Leftarrow P}{\Gamma \vdash \lceil p \rceil \Leftarrow \uparrow P} \qquad \textit{\textbf{pi}}\ \frac{\Gamma \vdash t \Leftarrow N}{\Gamma \vDash \lfloor t \rfloor \Leftarrow \downarrow N}$$

$$\textit{\textbf{ie}}\ \frac{\Gamma \vdash t \Rightarrow P \supset N \quad \Gamma \vDash p \Leftarrow P}{\Gamma \vdash t\ p \Rightarrow N} \qquad \textit{\textbf{ii}}\ \frac{\Gamma, x : P \vdash t \Leftarrow N}{\Gamma \vdash \lambda x.t \Leftarrow P \supset N}$$

$$\textit{\textbf{ne}}\ \frac{\Gamma \vdash u \Rightarrow \uparrow P \quad \Gamma, x : P \vdash t \Leftarrow M}{\Gamma \vdash u \;\mathsf{to}\; x.t \Leftarrow M} \qquad \textit{\textbf{pe}}\ \frac{\Gamma \vDash p \Rightarrow \downarrow N}{\Gamma \vdash \mathsf{F}\,p \Rightarrow N}$$

....

$$\textit{\textbf{mt}}\ \frac{\Gamma \vdash t \Rightarrow N \quad N \in \{a^-, \uparrow P\}}{\Gamma \vdash t \Leftarrow N} \qquad \textit{\textbf{ct}}\ \frac{\Gamma \vdash t \Leftarrow N}{\Gamma \vdash t \Rightarrow N}$$

$$\textit{\textbf{mp}}\ \frac{\Gamma \vDash p \Rightarrow P \quad P \in \{a^+, \downarrow N\}}{\Gamma \vDash p \Leftarrow P} \qquad \textit{\textbf{cp}}\ \frac{\Gamma \vDash p \Leftarrow P}{\Gamma \vDash p \Rightarrow P}$$

Figure 6. *Rules for bidirectional **NJPV** with associated terms*

3. Polarity $\overset{F}{\uparrow} / \overset{U}{\downarrow}$ are adjunctions $F \dashv U$,

but not the <u>same</u> as the modal adjunctions

seen in LNL/adjoint logic.

3. Polarity $\overset{F}{\uparrow}/\overset{U}{\downarrow}$ are adjunctions $F \dashv U$,

but not the <u>same</u> as the modal adjunctions seen in LNL/adjoint logic.

$$F : \text{Pers} \to \text{Lin}$$
$$U : \text{Lin} \to \text{Pers}$$

these preserve polarity!

$$! \triangleq F \downarrow U$$

⚠ WARNING: Notation $\downarrow\uparrow$ in adjoint logic uses the reverse convention for F vs. U!

Conclusion

Life lessons from proof theory

when you're struggling
to focus on what matters

Make the easy choices
before the difficult ones.

It's ok to do one thing at a time.

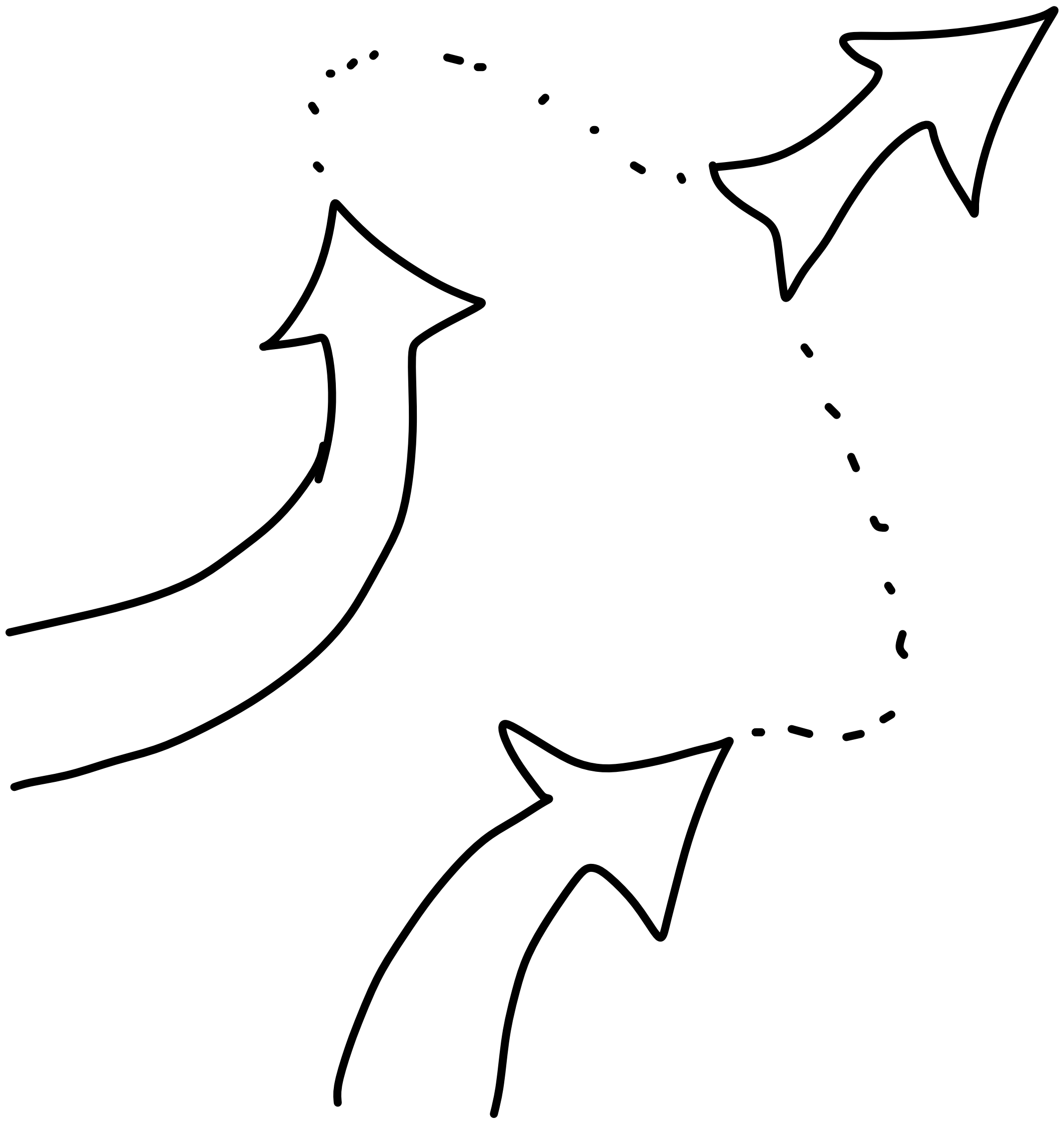
It's ok to do one thing at a time.

maybe even a good idea.

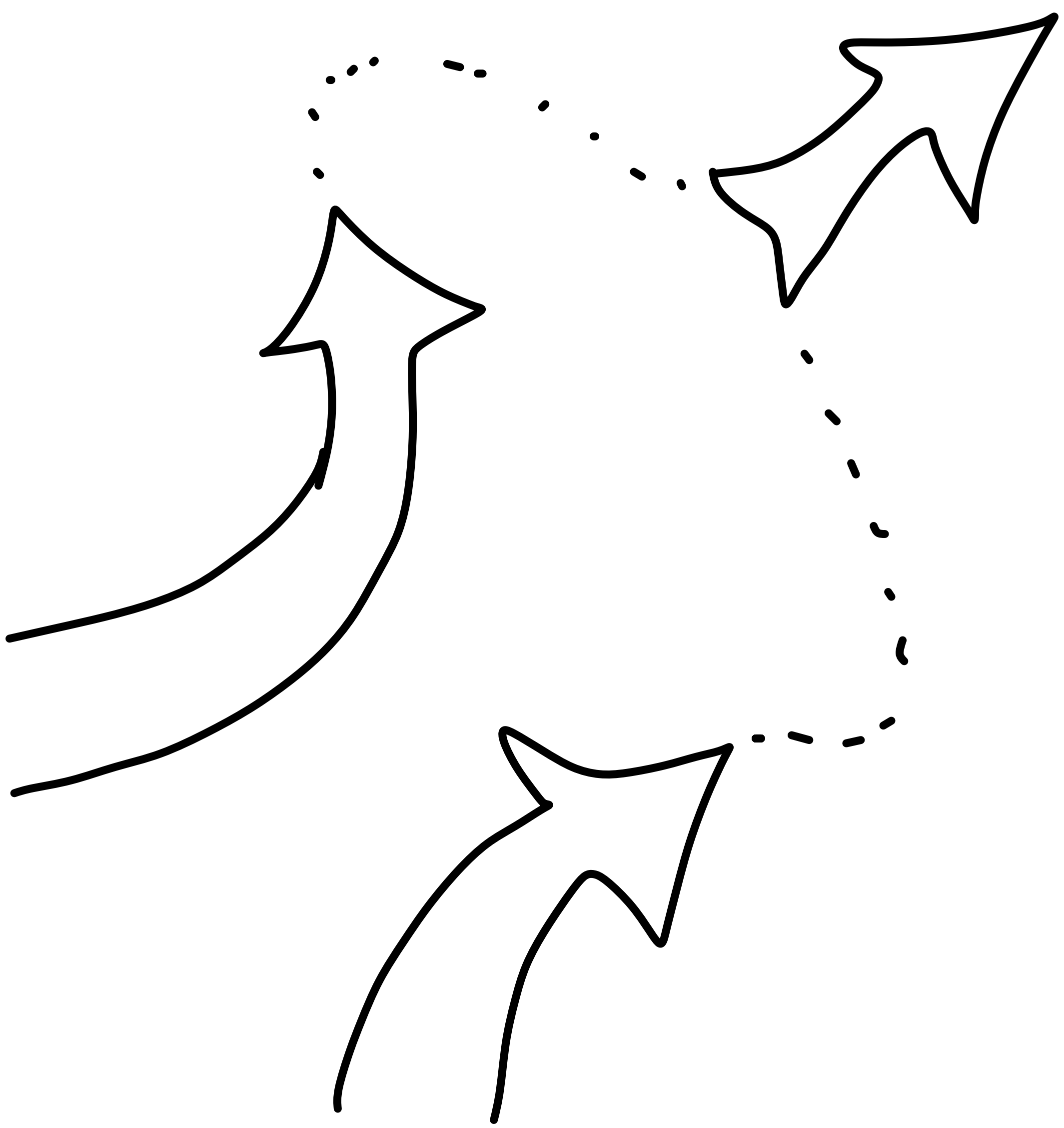
If you get stuck trying to reach your goal, If you get stuck trying to reach your goal,

try taking a closer look at what you already have.

Convergence may be a sign of significance.



Convergence



may be a sign of significance.

(but look carefully at the fine print.)

There's still more to do!

- Fully exploring the possibility space of orthogonal concepts
- Understanding relationships to categorical semantics
- Language implementation
- ITP integration
- Pedagogy

A guided tour of polarity and focusing

Chris Martens

- https://khoury.northeastern.edu/~cmartens/
- c.martens@northeastern.edu

- https://chrisamaphone.hyperkind.org/types-2025.html

References

- “Focusing on pattern matching.” Neel Krishnaswami, POPL 2009.
- “Structural Focalization.” Robert J. Simmons, TOCL 2014.
- “Computation in focused intuitionistic logic.” Taus Brock-Nannestad, Nicolas Guenot, Daniel Gustafsson. PPDP 2015.
- “Bidirectional Typing.” Jana Dunfield and Neel Krishnaswami. ACM Computing Surveys, 2021.
- “Call-by-push-value: A subsuming paradigm.” Paul Blain Levy. International Conference on Typed Lambda Calculi and Applications, 1999.
- "Logic programming with focusing proofs in linear logic.” Jean-Marc Andreolli, J. Logic and Computation, 1992.