# *A unified display proof theory for bunched logic*

James Brotherston

Imperial College London

MFPS 2010
University of Ottawa, 9 May 2010

# *Substructural logics: an overview*

Substructural logics restrict the structural principles of ordinary classical logic *(weakening, contraction, associativity, exchange...)*.

Examples:

# *Substructural logics: an overview*

Substructural logics restrict the structural principles of ordinary classical logic *(weakening, contraction, associativity, exchange. . . )*.

Examples:

- Lambek calculus totally rejects weakening and contraction (commutativity and associativity are optional too);

# *Substructural logics: an overview*

Substructural logics restrict the structural principles of ordinary classical logic *(weakening, contraction, associativity, exchange. . . )*.

Examples:

- Lambek calculus totally rejects weakening and contraction (commutativity and associativity are optional too);
- Linear logic permits weakening and contraction only for formulas prefixed with “exponential” modalities;

# *Substructural logics: an overview*

Substructural logics restrict the structural principles of ordinary classical logic *(weakening, contraction, associativity, exchange. . . )*.

Examples:

- Lambek calculus totally rejects weakening and contraction (commutativity and associativity are optional too);
- Linear logic permits weakening and contraction only for formulas prefixed with "exponential" modalities;
- Relevant logic replaces some of the standard 'additive' connectives, which obey weakening and contraction, with 'multiplicative' variants which do not;

# *Substructural logics: an overview*

Substructural logics restrict the structural principles of ordinary classical logic *(weakening, contraction, associativity, exchange...)*.

Examples:

- Lambek calculus totally rejects weakening and contraction (commutativity and associativity are optional too);
- Linear logic permits weakening and contraction only for formulas prefixed with "exponential" modalities;
- Relevant logic replaces some of the standard 'additive' connectives, which obey weakening and contraction, with 'multiplicative' variants which do not;
- Bunched logic is like relevant logic, but retains the additive connectives which relevant logic throws away on philosophical grounds (e.g. $\rightarrow$).

# *Motivation for bunched logic*

- So, bunched logics are essentially obtained by "splicing" an additive propositional logic with a multiplicative one.

# *Motivation for bunched logic*

- So, bunched logics are essentially obtained by "splicing" an additive propositional logic with a multiplicative one.
- This gives a nice Kripke-style resource semantics: Additive connectives have their usual meaning, and multiplicatives denote resource composition properties:

$$
\begin{aligned}
r \models F_1 \wedge F_2 &\Leftrightarrow r \models F_1 \text{ and } r \models F_2 \\
r \models F_1 * F_2 &\Leftrightarrow r = r_1 \circ r_2 \text{ and } r_1 \models F_1 \text{ and } r_2 \models F_2
\end{aligned}
$$

(where $\circ$ is a binary monoid operation).

# *Motivation for bunched logic*

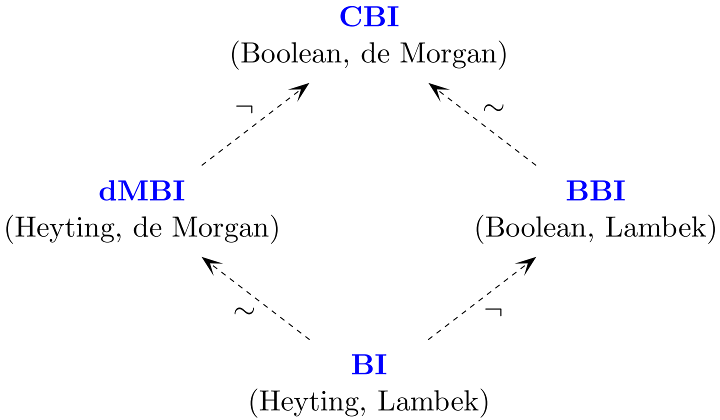
- So, bunched logics are essentially obtained by "splicing" an additive propositional logic with a multiplicative one.
- This gives a nice Kripke-style resource semantics: Additive connectives have their usual meaning, and multiplicatives denote resource composition properties:

$$
\begin{aligned}
r \models F_1 \wedge F_2 &\;\Leftrightarrow\; r \models F_1 \text{ and } r \models F_2 \\
r \models F_1 * F_2 &\;\Leftrightarrow\; r = r_1 \circ r_2 \text{ and } r_1 \models F_1 \text{ and } r_2 \models F_2
\end{aligned}
$$

  (where $\circ$ is a binary monoid operation).
- Taking particular models gives us separation logic and other spatial logics (used in program verification).

# *The bunched logic family*

Additives / multiplicatives can be classical or intuitionistic:




- Subtitles (X,Y) indicate the underlying algebras.
- Arrows denote addition of classical negations $\neg$ or $\sim$.

# *Bunched logics via elementary logics*

$$\begin{array}{lcccccc} \textit{Additives:} & \top & \bot & \neg & \vee & \wedge & \rightarrow \\ \textit{Multiplicatives:} & \top^* & \bot^* & \sim & \overset{*}{\vee} & * & -\!\!* \end{array}$$

- IL and CL are standard intuitionistic / classical logic over the additives;

# *Bunched logics via elementary logics*

| | | | | | | |
|---|---|---|---|---|---|---|
| *Additives:* | $\top$ | $\bot$ | $\neg$ | $\vee$ | $\wedge$ | $\rightarrow$ |
| *Multiplicatives:* | $\top^*$ | $\bot^*$ | $\sim$ | $\overset{*}{\vee}$ | $*$ | $-\!\!*$ |

- IL and CL are standard intuitionistic / classical logic over the additives;
- LM and dMM are (commutative and associative) Lambek / de Morgan logic over the multiplicatives;

# *Bunched logics via elementary logics*

| | | | | | | |
|---|---|---|---|---|---|---|
| *Additives:* | $\top$ | $\bot$ | $\neg$ | $\vee$ | $\wedge$ | $\rightarrow$ |
| *Multiplicatives:* | $\top^*$ | $\bot^*$ | $\sim$ | $\overset{*}{\vee}$ | $*$ | $-\!\!*$ |

- IL and CL are standard intuitionistic / classical logic over the additives;
- LM and dMM are (commutative and associative) Lambek / de Morgan logic over the multiplicatives;
- Define:

$$\begin{aligned} \mathrm{BI} &= \mathrm{IL} + \mathrm{LM} \\ \mathrm{BBI} &= \mathrm{CL} + \mathrm{LM} \\ \mathrm{dMBI} &= \mathrm{IL} + \mathrm{dMM} \\ \mathrm{CBI} &= \mathrm{CL} + \mathrm{dMM} \end{aligned}$$

  where + is union of minimal proof systems for the logics.

# LBI: *the* BI *sequent calculus*

- Sequents are $\Gamma \vdash F$ where $F$ a formula and $\Gamma$ a bunch:

$$\Gamma ::= F \mid \emptyset \mid \varnothing \mid \Gamma\,;\,\Gamma \mid \Gamma\,,\,\Gamma$$

# LBI: *the* BI *sequent calculus*

- Sequents are $\Gamma \vdash F$ where $F$ a formula and $\Gamma$ a bunch:

$$\Gamma ::= F \mid \emptyset \mid \varnothing \mid \Gamma\,;\,\Gamma \mid \Gamma\,,\,\Gamma$$

- Rules for $-\!\!*$ are:

$$\frac{\Delta \vdash F_1 \quad \Gamma(F_2) \vdash F}{\Gamma(\Delta\,,\,F_1 -\!\!* F_2) \vdash F}\,(-\!\!*\mathrm{L}) \qquad \frac{\Gamma\,,\,F \vdash G}{\Gamma \vdash F -\!\!* G}\,(-\!\!*\mathrm{R})$$

where $\Gamma(\Delta)$ is bunch $\Gamma$ with sub-bunch $\Delta$;

# LBI: *the* BI *sequent calculus*

- Sequents are $\Gamma \vdash F$ where $F$ a formula and $\Gamma$ a bunch:

$$\Gamma ::= F \mid \emptyset \mid \varnothing \mid \Gamma \,;\, \Gamma \mid \Gamma \,,\, \Gamma$$

- Rules for $-\!\!*$ are:

$$\frac{\Delta \vdash F_1 \quad \Gamma(F_2) \vdash F}{\Gamma(\Delta \,,\, F_1 -\!\!* F_2) \vdash F}\,(-\!\!*\mathrm{L}) \qquad\qquad \frac{\Gamma \,,\, F \vdash G}{\Gamma \vdash F -\!\!* G}\,(-\!\!*\mathrm{R})$$

  where $\Gamma(\Delta)$ is bunch $\Gamma$ with sub-bunch $\Delta$;

- LBI satisfies cut-elimination (Pym '02).
- Unfortunately cut-elimination breaks if we try to extend LBI to BBI, dMBI, CBI in the obvious way.

# *Display calculus: an overview*

- Display calculi manipulate consecutions $X \vdash Y$, with left- and right-introduction rules for each logical connective.

## *Display calculus: an overview*

- Display calculi manipulate consecutions $X \vdash Y$, with left- and right-introduction rules for each logical connective.
- Structures $X$ and $Y$ are built from formulas and structural connectives. Substructures of $X \vdash Y$ are classified as antecedent or consequent parts.

# *Display calculus: an overview*

- Display calculi manipulate consecutions $X \vdash Y$, with left- and right-introduction rules for each logical connective.
- Structures $X$ and $Y$ are built from formulas and structural connectives. Substructures of $X \vdash Y$ are classified as antecedent or consequent parts.
- In display calculi, one can rearrange consecutions:

*Definition*

$\equiv_D$ is a display-equivalence if for any antecedent (consequent) part $Z$ of $X \vdash Y$ we have $X \vdash Y \equiv_D Z \vdash W \quad (W \vdash Z)$.

## *Display calculus: an overview*

- Display calculi manipulate consecutions $X \vdash Y$, with left- and right-introduction rules for each logical connective.
- Structures $X$ and $Y$ are built from formulas and structural connectives. Substructures of $X \vdash Y$ are classified as antecedent or consequent parts.
- In display calculi, one can rearrange consecutions:

*Definition*

$\equiv_D$ is a display-equivalence if for any antecedent (consequent) part $Z$ of $X \vdash Y$ we have $X \vdash Y \equiv_D Z \vdash W \quad (W \vdash Z)$.

- Belnap '82 gives a set of syntactic conditions for display calculi which guarantee cut-elimination.

# *Display calculus: syntax*

- Structures are constructed from formulas and structural connectives:

| *Additive* | *Multiplicative* | *Arity* | *Antecedent* | *Consequent* |
|---|---|---|---|---|
| $\emptyset$ | $\varnothing$ | 0 | truth | falsity |
| $\sharp$ | $\flat$ | 1 | negation | negation |
| ; | , | 2 | conjunction | disjunction |
| $\Rightarrow$ | $\multimap$ | 2 | – | implication |

- Antecedent / consequent parts of consecutions $X \vdash Y$ are similar to positive / negative occurrences in formulas.

# *Display calculus: syntax*

- Structures are constructed from formulas and structural connectives:

| *Additive* | *Multiplicative* | *Arity* | *Antecedent* | *Consequent* |
|---|---|---|---|---|
| $\emptyset$ | $\varnothing$ | 0 | truth | falsity |
| $\sharp$ | $\flat$ | 1 | negation | negation |
| ; | , | 2 | conjunction | disjunction |
| $\Rightarrow$ | $\multimap$ | 2 | – | implication |

- Antecedent / consequent parts of consecutions $X \vdash Y$ are similar to positive / negative occurrences in formulas.
- We give display calculi for IL, CL, LM and dMM. Form of antecedent and consequent parts is restricted in each case.

# $\mathrm{DL_{CL}}$: *a display calculus for* CL

**Antecedent connectives:** $\emptyset \quad \sharp \quad ;$

**Consequent connectives:** $\emptyset \quad \sharp \quad ;$

**Display postulates:**

$$\begin{array}{ccccc} X ; Y \vdash Z & <>_D & X \vdash \sharp Y ; Z & <>_D & Y ; X \vdash Z \\ X \vdash Y ; Z & <>_D & X ; \sharp Y \vdash Z & <>_D & X \vdash Z ; Y \\ X \vdash Y & <>_D & \sharp Y \vdash \sharp X & <>_D & \sharp\sharp X \vdash Y \end{array}$$

**Logical rules:**

$$\frac{F \vdash X \quad G \vdash X}{F \vee G \vdash X}\,(\vee\mathrm{L}) \qquad \frac{X \vdash F_1 ; F_2}{X \vdash F_1 \vee F_2}\,(\vee\mathrm{R}) \qquad \text{(etc.)}$$

**Structural rules:**

$$\frac{\emptyset ; X \vdash Y}{X \vdash Y}\,(\emptyset\mathrm{L}) \qquad \frac{X \vdash Z}{X ; Y \vdash Z}\,(\mathrm{WkL}) \qquad \text{(etc.)}$$

## $\mathrm{DL_{LM}}$: *a display calculus for* LM

**Antecedent connectives:** $\varnothing \quad ,$

**Consequent connectives:** $\multimap$

**Display postulates:** $X \,,\, Y \vdash Z \;\; <>_D \;\; X \vdash Y \multimap Z \;\; <>_D \;\; Y \,,\, X \vdash Z$

**Logical rules:**

$$\frac{X \vdash F \quad G \vdash Y}{F \mathrel{-\!\!*} G \vdash X \multimap Y}\,(\mathrel{-\!\!*}\mathrm{L}) \qquad \frac{X \vdash F \multimap G}{X \vdash F \mathrel{-\!\!*} G}\,(\mathrel{-\!\!*}\mathrm{R}) \qquad \text{(etc.)}$$

**Structural rules:**

$$\frac{\varnothing \,,\, X \vdash Y}{X \vdash Y}\,(\varnothing\mathrm{L}) \qquad \frac{W \,,\, (X \,,\, Y) \vdash Z}{(W \,,\, X) \,,\, Y \vdash Z}\,(\mathrm{MAL})$$

## *Display calculi for bunched logics*

We obtain display calculi $\mathrm{DL}_{\mathcal{L}}$ for $\mathcal{L} \in \{\mathrm{BI}, \mathrm{BBI}, \mathrm{dMBI}, \mathrm{CBI}\}$ by:

$$\mathrm{DL}_{\mathcal{L}_1+\mathcal{L}_2} = \mathrm{DL}_{\mathcal{L}_1} + \mathrm{DL}_{\mathcal{L}_2}$$

where + is component-wise union of specifications.
The following hold for all our calculi:

# *Display calculi for bunched logics*

We obtain display calculi $\mathrm{DL}_{\mathcal{L}}$ for $\mathcal{L} \in \{\mathrm{BI}, \mathrm{BBI}, \mathrm{dMBI}, \mathrm{CBI}\}$ by:

$$\mathrm{DL}_{\mathcal{L}_1+\mathcal{L}_2} = \mathrm{DL}_{\mathcal{L}_1} + \mathrm{DL}_{\mathcal{L}_2}$$

where $+$ is component-wise union of specifications.
The following hold for all our calculi:

*Proposition (Display)*

*$\equiv_D$, given by the display postulates of $\mathrm{DL}_{\mathcal{L}}$, is indeed a display-equivalence for $\mathrm{DL}_{\mathcal{L}}$.*

## *Display calculi for bunched logics*

We obtain display calculi $\mathrm{DL}_{\mathcal{L}}$ for $\mathcal{L} \in \{\mathrm{BI}, \mathrm{BBI}, \mathrm{dMBI}, \mathrm{CBI}\}$ by:

$$\mathrm{DL}_{\mathcal{L}_1+\mathcal{L}_2} = \mathrm{DL}_{\mathcal{L}_1} + \mathrm{DL}_{\mathcal{L}_2}$$

where + is component-wise union of specifications.
The following hold for all our calculi:

*Proposition (Display)*

*$\equiv_D$, given by the display postulates of $\mathrm{DL}_{\mathcal{L}}$, is indeed a display-equivalence for $\mathrm{DL}_{\mathcal{L}}$.*

*Theorem (Soundness / Completeness)*

*$X \vdash Y$ is $\mathrm{DL}_{\mathcal{L}}$-provable iff its formula translation is provable in the minimal proof system for $\mathcal{L}$.*

# *Display calculi for bunched logics*

We obtain display calculi $\mathrm{DL}_{\mathcal{L}}$ for $\mathcal{L} \in \{\mathrm{BI}, \mathrm{BBI}, \mathrm{dMBI}, \mathrm{CBI}\}$ by:

$$\mathrm{DL}_{\mathcal{L}_1+\mathcal{L}_2} = \mathrm{DL}_{\mathcal{L}_1} + \mathrm{DL}_{\mathcal{L}_2}$$

where $+$ is component-wise union of specifications.
The following hold for all our calculi:

*Proposition (Display)*

*$\equiv_D$, given by the display postulates of $\mathrm{DL}_{\mathcal{L}}$, is indeed a display-equivalence for $\mathrm{DL}_{\mathcal{L}}$.*

*Theorem (Soundness / Completeness)*

*$X \vdash Y$ is $\mathrm{DL}_{\mathcal{L}}$-provable iff its formula translation is provable in the minimal proof system for $\mathcal{L}$.*

*Theorem (Cut-elimination)*

*Any $\mathrm{DL}_{\mathcal{L}}$ proof of $X \vdash Y$ can be algorithmically transformed into a cut-free $\mathrm{DL}_{\mathcal{L}}$ proof of $X \vdash Y$.*

## *Translating* LBI *into* $\mathrm{DL_{BI}}$

Recall the LBI rules for $\multimap$:

$$\frac{\Delta \vdash F_1 \quad \Gamma(F_2) \vdash F}{\Gamma(\Delta\ ,\ F_1 \multimap F_2) \vdash F}\,(\multimap\mathrm{L}) \qquad\qquad \frac{\Gamma\ ,\ F \vdash G}{\Gamma \vdash F \multimap G}\,(\multimap\mathrm{R})$$

$(\multimap\mathrm{R})$ has a direct equivalent in $\mathrm{DL_{BI}}$, while $(\multimap\mathrm{L})$ can be derived in $\mathrm{DL_{BI}}$ as follows:

# *Translating* LBI *into* $\mathrm{DL}_{\mathrm{BI}}$

Recall the LBI rules for $-\!\!*$:

$$\frac{\Delta \vdash F_1 \quad \Gamma(F_2) \vdash F}{\Gamma(\Delta \,,\, F_1 -\!\!* F_2) \vdash F}\,(-\!\!*\mathrm{L}) \qquad\qquad \frac{\Gamma \,,\, F \vdash G}{\Gamma \vdash F -\!\!* G}\,(-\!\!*\mathrm{R})$$

$(-\!\!*\mathrm{R})$ has a direct equivalent in $\mathrm{DL}_{\mathrm{BI}}$, while $(-\!\!*\mathrm{L})$ can be derived in $\mathrm{DL}_{\mathrm{BI}}$ as follows:

$$\frac{\dfrac{\Delta \vdash F_1 \qquad \dfrac{\Gamma(F_2) \vdash F}{F_2 \vdash X}\,(\mathrm{D}\equiv)}{\Delta \,,\, F_1 -\!\!* F_2 \vdash X}\,(-\!\!*\mathrm{L})}{\Gamma(\Delta \,,\, F_1 -\!\!* F_2) \vdash F}\,(\mathrm{D}\equiv)$$

Translation preserves cut-freeness of proofs.

# *Translating* $\mathrm{DL_{BI}}$ *into* LBI

For any $\mathrm{DL_{BI}}$ consecution $X \vdash Y$ define $\ulcorner X \vdash Y \urcorner$ as the result of maximally applying transformations:

$$\begin{array}{rcl} X \vdash Y \Rightarrow Z & \mapsto & X \,;\, Y \vdash Z \\ X \vdash Y \multimap Z & \mapsto & X \,,\, Y \vdash Z \end{array}$$

Note $\ulcorner X \vdash Y \urcorner$ is always an LBI sequent.

# *Translating* $\mathrm{DL_{BI}}$ *into* LBI

For any $\mathrm{DL_{BI}}$ consecution $X \vdash Y$ define $\ulcorner X \vdash Y \urcorner$ as the result of maximally applying transformations:

$$\begin{array}{rcl} X \vdash Y \Rightarrow Z & \mapsto & X \,;\, Y \vdash Z \\ X \vdash Y \multimap Z & \mapsto & X \,,\, Y \vdash Z \end{array}$$

Note $\ulcorner X \vdash Y \urcorner$ is always an LBI sequent.

Then the rules of $\mathrm{DL_{BI}}$ are LBI-derivable under $\ulcorner - \urcorner$, e.g.:

$$\frac{\ulcorner X \vdash F \urcorner \quad \ulcorner G \vdash Y \urcorner}{\ulcorner X \,,\, F \mathrel{-\!\!*} G \vdash Y \urcorner} \quad = \quad \frac{X \vdash F \quad \Gamma(G) \vdash H}{\Gamma(X \,,\, F \mathrel{-\!\!*} G) \vdash H}$$

Translation again preserves cut-freeness of proofs.

# *Display calculi vs. sequent calculi*

- By the two previous translations we have:

*Proposition*

*There is a one-to-many correspondence between cut-free proofs in* LBI *and cut-free proofs in* $\mathrm{DL}_{\mathrm{BI}}$.

So LBI can be seen as an optimised $\mathrm{DL}_{\mathrm{BI}}$.

# *Display calculi vs. sequent calculi*

- By the two previous translations we have:

*Proposition*

*There is a one-to-many correspondence between cut-free proofs in* LBI *and cut-free proofs in* $\mathrm{DL}_{\mathrm{BI}}$.

So LBI can be seen as an optimised $\mathrm{DL}_{\mathrm{BI}}$.

- However, display proofs for BBI, dMBI, CBI do not easily translate to sequent proofs in the same way. E.g., it is not obvious how to translate the $\mathrm{DL}_{\mathrm{BBI}}$ consecution $F \,,\, \sharp G \vdash H$ into a sequent without the unary $\sharp$.

# *Display calculi vs. sequent calculi*

- By the two previous translations we have:

*Proposition*

*There is a one-to-many correspondence between cut-free proofs in* LBI *and cut-free proofs in* $\mathrm{DL_{BI}}$.

So LBI can be seen as an optimised $\mathrm{DL_{BI}}$.

- However, display proofs for BBI, dMBI, CBI do not easily translate to sequent proofs in the same way. E.g., it is not obvious how to translate the $\mathrm{DL_{BBI}}$ consecution $F \,, \sharp G \vdash H$ into a sequent without the unary $\sharp$.
- Thus we claim that our display calculi really are canonical proof systems for the bunched logics.

# *Applications*

- Cut-free proof search is still very difficult (display rules, structural rules).

# *Applications*

- Cut-free proof search is still very difficult (display rules, structural rules).
- In general, for both display and sequent calculi:

$$\text{cut-elimination} \not\Rightarrow \text{(semi)decidability}$$

*(cf. linear logic, relevant logic, arithmetic ...)*

# *Applications*

- Cut-free proof search is still very difficult (display rules, structural rules).
- In general, for both display and sequent calculi:

$$\text{cut-elimination} \not\Rightarrow \text{(semi)decidability}$$

*(cf. linear logic, relevant logic, arithmetic ...)*

- Indeed, while BI is known decidable (Galmiche et al. '05), BBI and CBI are known undecidable (Brotherston and Kanovich '10, Larchey-Wendling and Galmiche '10).

# *Applications*

- Cut-free proof search is still very difficult (display rules, structural rules).
- In general, for both display and sequent calculi:

$$\text{cut-elimination} \not\Rightarrow \text{(semi)decidability}$$

*(cf. linear logic, relevant logic, arithmetic ...)*

- Indeed, while BI is known decidable (Galmiche et al. '05), BBI and CBI are known undecidable (Brotherston and Kanovich '10, Larchey-Wendling and Galmiche '10).
- Cut-elimination provides structure and removes infinite branching points from the proof search space.

# *Applications*

- Cut-free proof search is still very difficult (display rules, structural rules).
- In general, for both display and sequent calculi:

  cut-elimination $\not\Rightarrow$ (semi)decidability

  *(cf. linear logic, relevant logic, arithmetic . . . )*
- Indeed, while BI is known decidable (Galmiche et al. '05), BBI and CBI are known undecidable (Brotherston and Kanovich '10, Larchey-Wendling and Galmiche '10).
- Cut-elimination provides structure and removes infinite branching points from the proof search space.
- Our calculi could be potentially be used in interactive theorem proving tools (proof-by-pointing) or to define partial search strategies.