Logic Beyond Formulas: Designing Proof Systems on Graphs

Matteo Acclavio

ROMA TRE
UNIVERSITÀ DEGLI STUDI

Based in on joint works with Lutz Straßburger, Ross Horne and Sjouke Mauw

LATD2022 07/09/2022

Why at LATD?

Why at LATD?

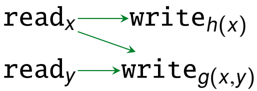
We invite contributions on any relevant aspects of logical systems (including many valued, fuzzy, substructural, modal and quantum logics), in particular:

- Proof theory and computational complexity
- Algebraic semantics and abstract algebraic logic
- First-order, higher-order and modal formalisms
- Geometric and game-theoretic aspects
- Applications and foundational issues

Definitively Related-to "-ish"

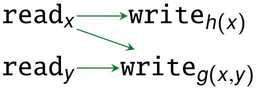
Logical Time

Happens-before relation is crucial in distributed systems



[1] Lamport '78

Happens-before relation is crucial in distributed systems

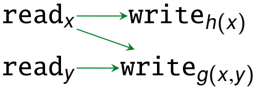


Logical Clocks[1] = Enforcing specific total orders on events

✓	$\mathtt{read}_x$	$\lessdot$	$\mathtt{write}_{h(x)}$	$\lessdot$	$\mathtt{read}_y$	$\lessdot$	$\mathtt{write}_{g(x,y)}$
✓	$\mathtt{read}_x$	$\lessdot$	$\mathtt{read}_y$	$\lessdot$	$\mathtt{write}_{g(x,y)}$	$\lessdot$	$\mathtt{write}_{h(x)}$
✓	$\mathtt{read}_x$	$\lessdot$	$\mathtt{read}_y$	$\lessdot$	$\mathtt{write}_{h(x)}$	$\lessdot$	$\mathtt{write}_{g(x,y)}$
✓	$\mathtt{read}_y$	$\lessdot$	$\mathtt{read}_x$	$\lessdot$	$\mathtt{write}_{g(x,y)}$	$\lessdot$	$\mathtt{write}_{h(x)}$
✓	$\mathtt{read}_y$	$\lessdot$	$\mathtt{read}_x$	$\lessdot$	$\mathtt{write}_{h(x)}$	$\lessdot$	$\mathtt{write}_{g(x,y)}$
✗	$\mathtt{read}_x$	$\lessdot$	$\mathtt{write}_{h(x)}$	$\lessdot$	$\mathtt{write}_{g(x,y)}$	$\lessdot$	$\mathtt{read}_y$

[1] Lamport ’78

Happens-before relation is crucial in distributed systems



Logical Clocks[1] = Enforcing specific total orders on events

✓	$\mathtt{read}_x \lessdot \mathtt{write}_{h(x)} \lessdot \mathtt{read}_y \lessdot \mathtt{write}_{g(x,y)}$
✓	$\mathtt{read}_x \lessdot \mathtt{read}_y \lessdot \mathtt{write}_{g(x,y)} \lessdot \mathtt{write}_{h(x)}$
✓	$\mathtt{read}_x \lessdot \mathtt{read}_y \lessdot \mathtt{write}_{h(x)} \lessdot \mathtt{write}_{g(x,y)}$
✓	$\mathtt{read}_y \lessdot \mathtt{read}_x \lessdot \mathtt{write}_{g(x,y)} \lessdot \mathtt{write}_{h(x)}$
✓	$\mathtt{read}_y \lessdot \mathtt{read}_x \lessdot \mathtt{write}_{h(x)} \lessdot \mathtt{write}_{g(x,y)}$
✗	$\mathtt{read}_x \lessdot \mathtt{write}_{h(x)} \lessdot \mathtt{write}_{g(x,y)} \lessdot \mathtt{read}_y$

Logical Time = Happens-before relation without clocks

[1] Lamport '78

Aim of this line of works:

Proof Theory treating the happens-before relation "logically"

Aim of this line of works:

Proof Theory treating the happens-before relation "logically"

That is:

Logical time is expressed by logical connectives

A "happens before" B $\rightsquigarrow$ $A \triangleleft B$

Aim of this line of works:

Proof Theory treating the happens-before relation "logically"

That is:

Logical time is expressed by logical connectives

$$A \text{ "happens before" } B \quad \rightsquigarrow \quad A \triangleleft B$$

and

Logical implication ($\multimap$) capturing partial order refinements

Why *Graphs*?

Previous attempts: Pomset logic

Pomset formulas

$$A, B ::= a \mid a^{\perp} \mid A \parr B \mid A \lhd B \mid A \otimes B$$

$\parr$	$\lhd$	$\otimes$
disjunction	happens-before	conjunction
parallelism	sequentiality	“independence”

Previous attempts: Pomset logic

Pomset formulas

$$A, B ::= a \mid a^{\perp} \mid A \parr B \mid A \lhd B \mid A \otimes B$$

$\parr$ disjunction	$\lhd$ happens-before	$\otimes$ conjunction
parallelism (commutative)	sequentiality (non-commutative)	“independence” (commutative)

Previous attempts: Pomset logic

Pomset formulas

$$A, B ::= a \mid a^{\perp} \mid A \parr B \mid A \lhd B \mid A \otimes B$$

$\parr$ disjunction	$\lhd$ happens-before	$\otimes$ conjunction
parallelism (commutative)	sequentiality (non-commutative)	"independence" (commutative)

Negation $(\cdot)^{\perp}$ such that:

$$A^{\perp\perp} = A$$

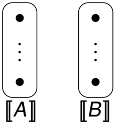
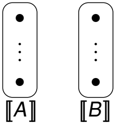
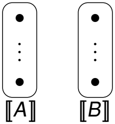
$$(A \parr B)^{\perp} = A^{\perp} \otimes B^{\perp} \qquad (A \lhd B)^{\perp} = A^{\perp} \lhd B^{\perp} \qquad (A \otimes B)^{\perp} = A^{\perp} \parr B^{\perp}$$

Implication defined "classically":

$$A \multimap B := A^{\perp} \parr B$$

Formulas and Graphs

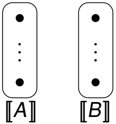
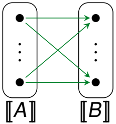
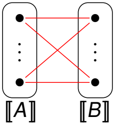
Relation webs[2] = graphs encoding Pomset formulas

$[\![a]\!]$	$[\![a^\perp]\!]$	$[\![A \parr B]\!] = [\![A]\!] \parr [\![B]\!]$	$[\![A \lhd B]\!] = [\![A]\!] \lhd [\![B]\!]$	$[\![A \otimes B]\!] = [\![A]\!] \otimes [\![B]\!]$
a	$a^\perp$			

[2] Retoré '93, Guglielmi '07

Formulas and Graphs

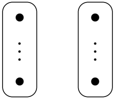
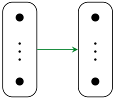
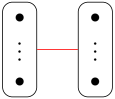
Relation webs[2] = graphs encoding Pomset formulas

$[\![a]\!]$	$[\![a^\perp]\!]$	$[\![A \parr B]\!] = [\![A]\!] \parr [\![B]\!]$	$[\![A \triangleleft B]\!] = [\![A]\!] \triangleleft [\![B]\!]$	$[\![A \otimes B]\!] = [\![A]\!] \otimes [\![B]\!]$
a	$a^\perp$			

[2] Retoré '93, Guglielmi '07

Formulas and Graphs

Relation webs[2] = graphs encoding Pomset formulas

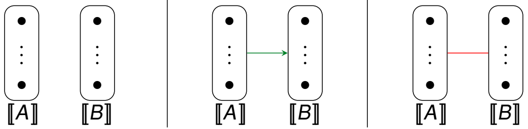
$[\![a]\!]$	$[\![a^\perp]\!]$	$[\![A \parr B]\!] = [\![A]\!] \parr [\![B]\!]$	$[\![A \lhd B]\!] = [\![A]\!] \lhd [\![B]\!]$	$[\![A \otimes B]\!] = [\![A]\!] \otimes [\![B]\!]$
a	$a^\perp$	 $[\![A]\!]$ $[\![B]\!]$	 $[\![A]\!]$ $[\![B]\!]$	 $[\![A]\!]$ $[\![B]\!]$

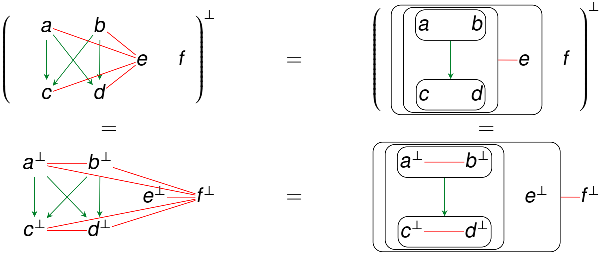
[2] Retoré '93, Guglielmi '07

Formulas and Graphs

Relation webs[2] = graphs encoding Pomset formulas

$[\![a]\!]$	$[\![a^\perp]\!]$	$[\![A \parr B]\!] = [\![A]\!] \parr [\![B]\!]$	$[\![A \lhd B]\!] = [\![A]\!] \lhd [\![B]\!]$	$[\![A \otimes B]\!] = [\![A]\!] \otimes [\![B]\!]$
a	$a^\perp$			

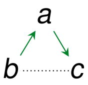




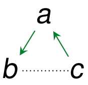
[2] Retoré '93, Guglielmi '07

Formulas and Graphs

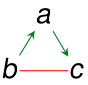
A graph containing an induced subgraph of the following shape cannot be represented by a formula:



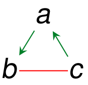
or

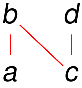


or

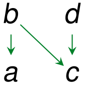


or

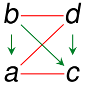


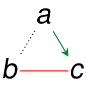


or

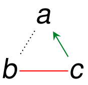


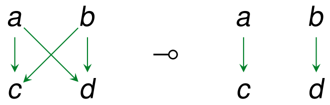
or





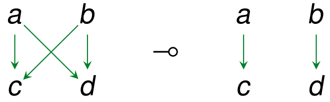
or





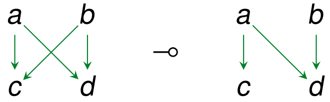
$$(a \parr b) \lhd (c \parr d) \multimap (a \lhd c) \parr (b \lhd d)$$

Provable in BV $\subset$ Pomset

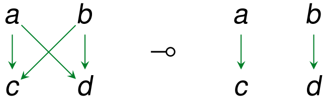


$(a \parr b) \triangleleft (c \parr d) \multimap (a \triangleleft c) \parr (b \triangleleft d)$

Provable in BV ⊂ Pomset

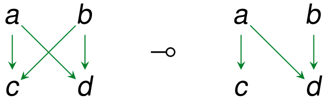


$(a \parr b) \triangleleft (c \parr d) \multimap$ NOT A FORMULA

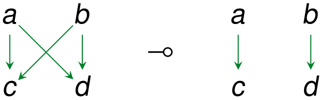


$$(a \parr b) \lhd (c \parr d) \multimap (a \lhd c) \parr (b \lhd d)$$

Provable in BV $\subset$ Pomset (series-parallel orders)

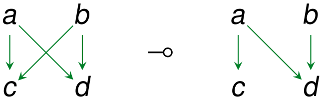


$(a \parr b) \lhd (c \parr d) \multimap$ NOT A FORMULA



$$(a \parr b) \triangleleft (c \parr d) \multimap (a \triangleleft c) \parr (b \triangleleft d)$$

Provable in BV $\subset$ Pomset
(series-parallel orders)



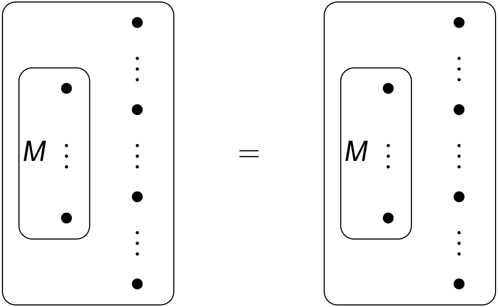
$(a \parr b) \triangleleft (c \parr d) \multimap$ NOT A FORMULA

Provable in $\mathrm{GV}^{\mathrm{sl}}$
(our result)

Preliminaries on Graphs

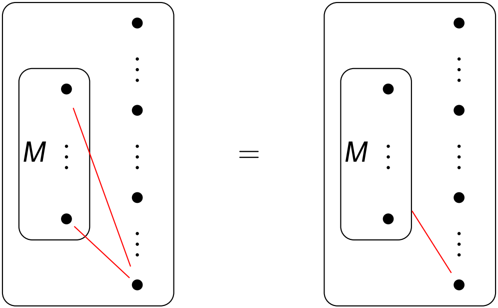
Modular Decomposition

A ***module*** of a graph $G = H[M]$ is a set of vertices M s.t.



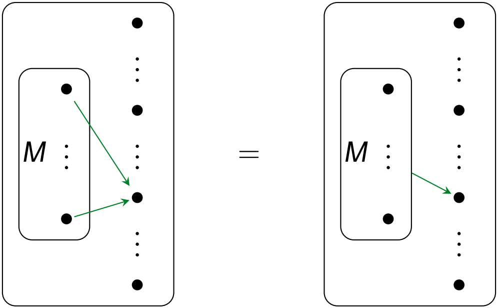
Modular Decomposition

A ***module*** of a graph $G = H[M]$ is a set of vertices M s.t.



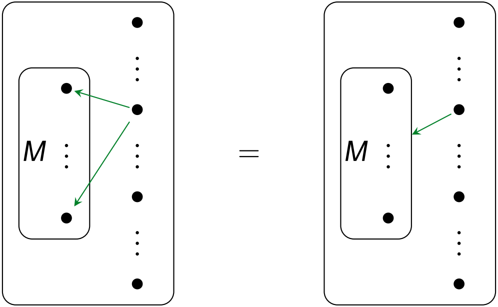
Modular Decomposition

A ***module*** of a graph $G = H[M]$ is a set of vertices M s.t.



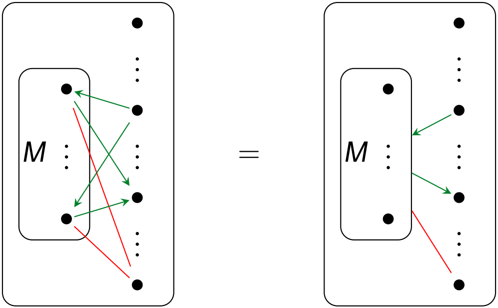
Modular Decomposition

A ***module*** of a graph $G = H[M]$ is a set of vertices M s.t.



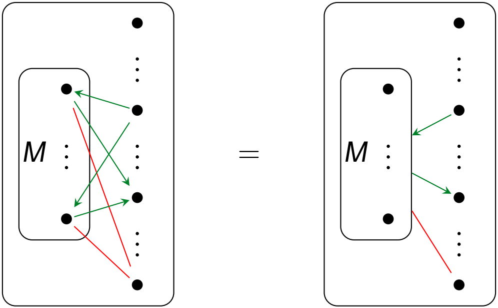
Modular Decomposition

A ***module*** of a graph $G = H[M]$ is a set of vertices M s.t.

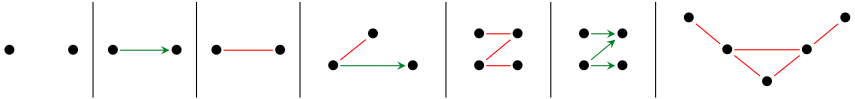


Modular Decomposition

A ***module*** of a graph $G = H[M]$ is a set of vertices M s.t.



A graph G is ***prime*** if it has modules V_G, $\emptyset$ and $\{x\}$ for all $x \in V_G$.



If G has n vertices and $H_1, \ldots, H_n$ graphs,
then we use G as a logic connective and we write $G(\!|H_1, \ldots, H_n|\!)$

$\parr$: 	$\triangleleft$: 	$\otimes$: 
$\parr(\!\|G, H\|\!) = G \parr H$	$\triangleleft(\!\|G, H\|\!) = G \triangleleft H$	$\otimes(\!\|G, H\|\!) = G \otimes H$

If G has n vertices and $H_1, \ldots, H_n$ graphs,
then we use G as a logic connective and we write $G(\!| H_1, \ldots, H_n |\!)$

$\parr$: 	$\triangleleft$: 	$\otimes$: 
$\parr(\!| G, H |\!) = G \parr H$	$\triangleleft(\!| G, H |\!) = G \triangleleft H$	$\otimes(\!| G, H |\!) = G \otimes H$

Lemma (Modular decomposition of graphs (Gallai '75))

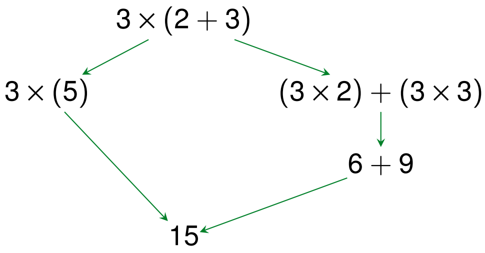
If $G \neq \emptyset$ is a graph, then we have exactly one of the following cases:

(i) *G is a singleton graph*

(ii) *$G = P(\!| A_1, \ldots, A_n |\!)$ for a prime graph P*

Graphical proof systems

On Deep Inference


3 × (2 + 3)
3 × (5)
(3 × 2) + (3 × 3)
6 + 9
15

$$\textit{mult}\ \dfrac{\dfrac{\dfrac{3\times(2+3)}{3\times\left(\textit{sum}\ \dfrac{2+3}{5}\right)}}{3\times 5}}{15}$$

$$\textit{dist}\ \dfrac{3\times(2+3)}{\dfrac{\left(\textit{mult}\ \dfrac{3\times 2}{6}\right)+\left(\textit{mult}\ \dfrac{3\times 3}{9}\right)}{\textit{sum}\ \dfrac{6+9}{15}}}$$

Deep inference does the same!

- $$\begin{array}{c} H_1 \\ \mathcal{D}_1 \Vert \\ G_1 \end{array} \quad \text{and} \quad \begin{array}{c} H_2 \\ \mathcal{D}_2 \Vert \\ G_2 \end{array} \quad \text{and} \quad \mathit{rule}\,\frac{G_1}{H_2} \quad \Longrightarrow \quad \mathit{rule}\,\frac{\begin{array}{c} H_1 \\ \mathcal{D}_1 \Vert \\ G_1 \end{array}}{\begin{array}{c} H_2 \\ \mathcal{D}_2 \Vert \\ G_2 \end{array}}$$

- $$\begin{array}{c} H_i \\ \mathcal{D}_i \Vert \\ G_i \end{array} \quad \text{and} \quad P \text{ an } n\text{-ary connective} \quad \Longrightarrow \quad P\left(\begin{array}{c} H_1 \\ \mathcal{D}_1 \Vert \\ G_1 \end{array}, \dots, \begin{array}{c} H_n \\ \mathcal{D}_n \Vert \\ G_n \end{array}\right)$$

The rules

$$\text{ai}\downarrow \frac{\emptyset}{a^\perp \parr a} \qquad \text{ai}\uparrow \frac{a^\perp \otimes a}{\emptyset}$$

$$\text{s}_{\parr} \frac{P(\!|M_1, \ldots, M_{i-1}, \mathbf{M_i} \parr N, M_{i+1}, \ldots M_n|\!)}{\mathbf{M_i} \parr P(\!|M_1, \ldots, M_{i-1}, N, M_{i+1}, \ldots, M_n|\!)} \qquad \text{s}_{\otimes} \frac{\mathbf{M_i} \otimes P(\!|M_1, \ldots, M_{i-1}, N, M_{i+1}, \ldots, M_n|\!)}{P(\!|M_1, \ldots, M_{i-1}, \mathbf{M_i} \otimes N, M_{i+1}, \ldots, M_n|\!)}$$

$$\text{p}\downarrow \frac{(\mathbf{M_1} \parr N_1) \otimes \cdots \otimes (\mathbf{M_n} \parr N_n)}{R^\perp(\!|\mathbf{M_1}, \ldots, \mathbf{M_n}|\!) \parr R(\!|N_1, \ldots, N_n|\!)} \qquad \text{p}\uparrow \frac{R(\!|\mathbf{M_1}, \ldots, \mathbf{M_n}|\!) \otimes R^\perp(\!|N_1, \ldots, N_n|\!)}{(\mathbf{M_1} \otimes N_1) \parr \cdots \parr (\mathbf{M_n} \otimes N_n)}$$

$$\text{q}\downarrow \frac{Q^\perp(\!|L_1 \parr J_1, \ldots, L_n \parr J_n|\!)}{Q^\perp(\!|L_1, \ldots, L_n|\!) \parr Q(\!|J_1, \ldots, J_n|\!)} \qquad \text{q}\uparrow \frac{Q(\!|L_1, \ldots, L_n|\!) \otimes Q^\perp(\!|J_1, \ldots, J_n|\!)}{Q(\!|L_1 \otimes J_1, \ldots, L_n \otimes J_n|\!)}$$

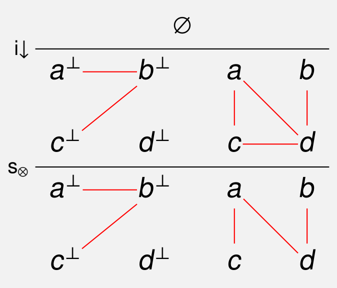
$$\text{qm} \frac{Q(\!|L_1 \parr J_1, \ldots, L_n \parr J_n|\!)}{Q(\!|L_1, \ldots, L_n|\!) \parr Q(\!|J_1, \ldots, J_n|\!)}$$

$$\text{sl} \frac{Q(\!|M_1, \ldots, M_k, \emptyset, \ldots, \emptyset|\!) \triangleleft Q(\!|\emptyset, \ldots, \emptyset, M_{k+1}, \ldots, M_n|\!)}{Q(\!|M_1, \ldots, M_n|\!)}$$

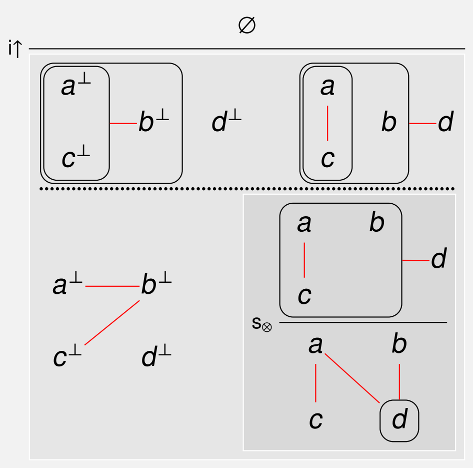
P, Q and R prime graphs with R red-white and Q green-white. M_i and $L_i \parr J_i$ are non-empty

Note: in this work graphs without three-color prime graphs in the modular decomposition

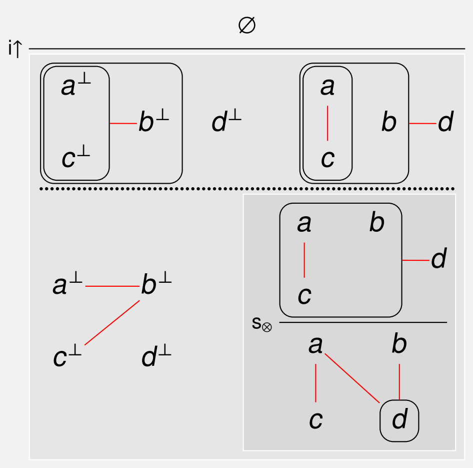
An example:



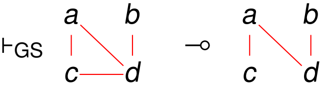
An example:



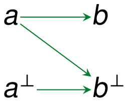
An example:



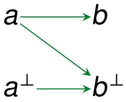
That is:



The rule sl refines a partial order *slicing* a "before" and an "after"

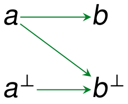


The rule sl refines a partial order *slicing* a “before” and an “after”

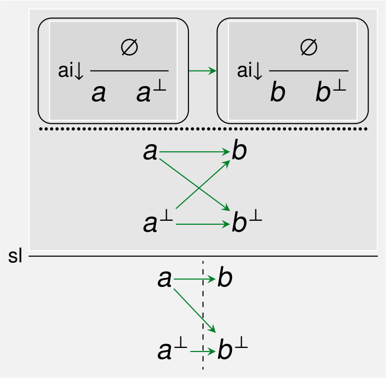


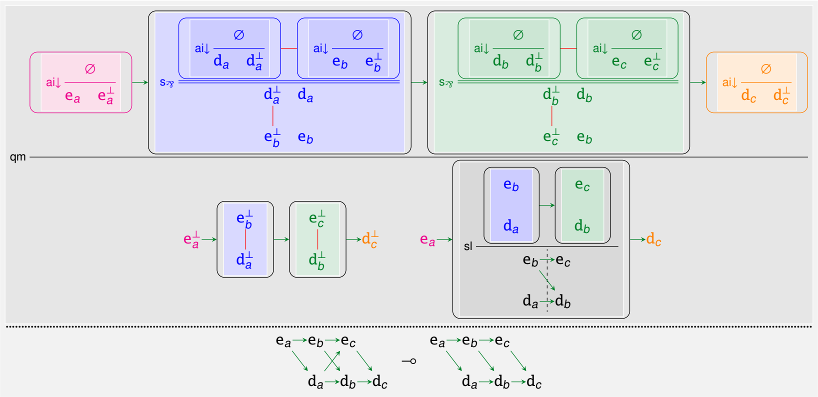
This rules implements Logical Clocks!

The rule sl refines a partial order *slicing* a "before" and an "after"

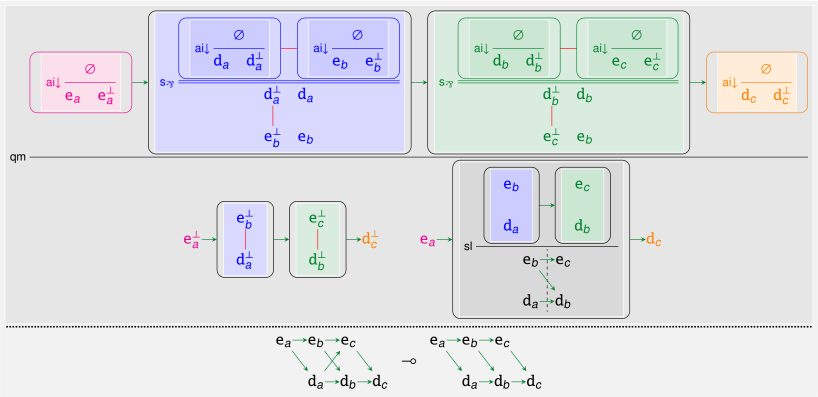


This rules implements Logical Clocks!





ai↓
∅
e_a $e_a^\perp$
s⅋
d_a $d_a^\perp$
e_b $e_b^\perp$
$d_a^\perp$ d_a
$e_b^\perp$ e_b
d_b $d_b^\perp$
e_c $e_c^\perp$
$d_b^\perp$ d_b
$e_c^\perp$ e_b
d_c $d_c^\perp$
qm
sl
$e_a \to e_b \to e_c$
$d_a \to d_b \to d_c$
⊸



We can proof theoretically prove that 2-Queues can simulate 3-Queues

Transitivity of $\multimap$

Remark

if $A \multimap B$ and $B \multimap C$, then $A \multimap C$

Remark

if $(A^\perp \parr B)$ and $(B^\perp \parr C)$, then $(A^\perp \parr C)$

Remark

if $(A^\perp \parr B) \otimes (B^\perp \parr C)$, then $(A^\perp \parr C)$

Remark

if $(A^\perp \parr B) \otimes (B^\perp \parr C)$, then $(A^\perp \parr C)$

$$\mathsf{p}{\downarrow}\, \frac{\begin{array}{c}\emptyset\\ \| \\ A^\perp \parr B\end{array} \otimes \begin{array}{c}\emptyset\\ \| \\ B^\perp \parr C\end{array}}{= \dfrac{A^\perp \parr \left(\mathsf{i}{\uparrow}\, \dfrac{B^\perp \otimes B}{\emptyset}\right) \parr C}{A^\perp \parr C}}$$

Remark

if $(A^\perp \parr B) \otimes (B^\perp \parr C)$, then $(A^\perp \parr C)$

$$
\mathsf{p}{\downarrow}\,\frac{\begin{array}{c}\emptyset\\ \Vert\\ A^\perp \parr B\end{array} \otimes \begin{array}{c}\emptyset\\ \Vert\\ B^\perp \parr C\end{array}}{=\,\dfrac{A^\perp \parr \left(\mathsf{i}{\uparrow}\,\dfrac{B^\perp \otimes B}{\emptyset}\right) \parr C}{A^\perp \parr C}}
$$

Cut Elimination = the rule $\mathsf{i}{\uparrow}\,\dfrac{A \otimes A^\perp}{\emptyset}$ is admissible

Theorem

The rule $\mathsf{i}\!\uparrow$ *is derivable in* $\{\mathsf{ai}\!\uparrow, \mathsf{p}\!\uparrow, \mathsf{q}\!\uparrow\}$

Theorem

- *rules* $\{\mathsf{ai}\!\uparrow, \mathsf{p}\!\uparrow, \mathsf{s}_{\otimes}\}$ *are admissible in* $\mathsf{GS} = \{\mathsf{ai}\!\downarrow, \mathsf{s}_{\invamp}, \mathsf{p}\!\downarrow\}$
- *rules* $\{\mathsf{ai}\!\uparrow, \mathsf{p}\!\uparrow, \mathsf{q}\!\uparrow\}$ *are admissible in* $\mathsf{GV} = \{\mathsf{ai}\!\downarrow, \mathsf{s}_{\invamp}, \mathsf{s}_{\otimes}, \mathsf{p}\!\downarrow, \mathsf{q}\!\downarrow, \mathsf{qm}\}$
- *rules* $\{\mathsf{ai}\!\uparrow, \mathsf{p}\!\uparrow, \mathsf{q}\!\uparrow\}$ *are admissible in* $\mathsf{GV}^{\mathsf{sl}} = \{\mathsf{ai}\!\downarrow, \mathsf{s}_{\invamp}, \mathsf{s}_{\otimes}, \mathsf{p}\!\downarrow, \mathsf{q}\!\downarrow, \mathsf{qm}, \mathsf{sl}\}$

$$\mathsf{ai}\!\downarrow \frac{\varnothing}{a^\perp \parr a} \qquad \mathsf{ai}\!\uparrow \frac{a^\perp \otimes a}{\varnothing}$$

$$\mathsf{s}_{\parr} \frac{P(\!|M_1, \dots, M_{i-1}, M_i \parr N, M_{i+1}, \dots M_n|\!)}{M_i \parr P(\!|M_1, \dots, M_{i-1}, N, M_{i+1}, \dots, M_n|\!)} \qquad \mathsf{s}_{\otimes} \frac{M_i \otimes P(\!|M_1, \dots, M_{i-1}, N, M_{i+1}, \dots, M_n|\!)}{P(\!|M_1, \dots, M_{i-1}, M_i \otimes N, M_{i+1}, \dots, M_n|\!)}$$

$$\mathsf{p}\!\downarrow \frac{(M_1 \parr N_1) \otimes \cdots \otimes (M_n \parr N_n)}{R^\perp(\!|M_1, \dots, M_n|\!) \parr R(\!|N_1, \dots, N_n|\!)} \qquad \mathsf{p}\!\uparrow \frac{R(\!|M_1, \dots, M_n|\!) \otimes R^\perp(\!|N_1, \dots, N_n|\!)}{(M_1 \otimes N_1) \parr \cdots \parr (M_n \otimes N_n)}$$

$$\mathsf{q}\!\downarrow \frac{Q^\perp(\!|L_1 \parr J_1, \dots, L_n \parr J_n|\!)}{Q^\perp(\!|L_1, \dots, L_n|\!) \parr Q(\!|J_1, \dots, J_n|\!)} \qquad \mathsf{q}\!\uparrow \frac{Q(\!|L_1, \dots, L_n|\!) \otimes Q^\perp(\!|J_1, \dots, J_n|\!)}{Q(\!|L_1 \otimes J_1, \dots, L_n \otimes J_n|\!)}$$

$$\mathsf{qm} \frac{Q(\!|L_1 \parr J_1, \dots, L_n \parr J_n|\!)}{Q(\!|L_1, \dots, L_n|\!) \parr Q(\!|J_1, \dots, J_n|\!)}$$

$$\mathsf{sl} \frac{Q(\!|M_1, \dots, M_k, \varnothing, \dots, \varnothing|\!) \triangleleft Q(\!|\varnothing, \dots, \varnothing, M_{k+1}, \dots, M_n|\!)}{Q(\!|M_1, \dots, M_n|\!)}$$

How to prove “cut-elimination”?

How to prove “cut-elimination”?

Splitting:

Pick a connected component of a provable $G \parr P(\!|M_1, \ldots, M_n|\!)$.
You can apply rules to G until you have a rule destroying P.

How to prove “cut-elimination”?

Splitting:

Pick a connected component of a provable $G \parr P(\!|M_1, \ldots, M_n|\!)$.
You can apply rules to G until you have a rule destroying P.

Context-Reduction:

You can apply splitting to a connector P deep in a context.

How to prove “cut-elimination”?

Splitting:

Pick a connected component of a provable $G \parr P\llparenthesis M_1, \ldots, M_n \rrparenthesis$.
You can apply rules to G until you have a rule destroying P.

Context-Reduction:

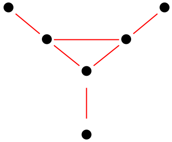
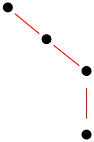
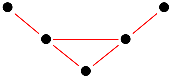
You can apply splitting to a connector P deep in a context.

Up-elimination:

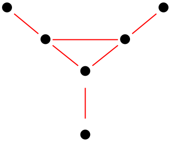
If the premise of an up-rule is provable, then its conclusion also is.

Conservativity

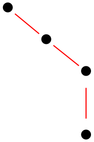
A new notion of *"sub-formula"* analyticity arises from this work:

 has sub-connective (e.g.)  and 

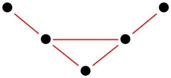
A new notion of *“sub-formula”* analyticity arises from this work:



has sub-connective (e.g.)

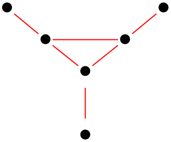
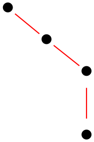
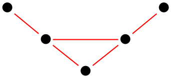


and



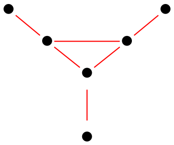
Proper sub-connective: sub-connective with $\frown \neq \emptyset$

A new notion of *"sub-formula"* analyticity arises from this work:

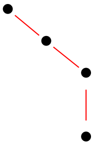
 has sub-connective (e.g.)  and 

Proper sub-connective: sub-connective with $\frown \neq \emptyset$
Analytic proof: only proper sub-connective of the conclusion

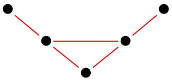
A new notion of *“sub-formula”* analyticity arises from this work:



has sub-connective (e.g.)



and



Proper sub-connective: sub-connective with $\frown \neq \emptyset$
Analytic proof: only proper sub-connective of the conclusion

Theorem

In GV *and* $\mathsf{GV}^{\mathsf{sl}}$*, if G is provable, than G admits an analytic proof.*

Conclusions and Future Works

Our results:

(-) GS, GV and GV^{sl} are proof systems [in the sense of Cook-Reckhow]

Our results:

(-) GS, GV and GV^{sl} are proof systems [in the sense of Cook-Reckhow]

(-) Size of a proof is polynomialy bounded

(-) GS is a conservative extension of MLL

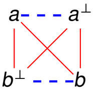
(-) GV and GV^{sl} are both conservative extensions of both BV and GS

Future works:

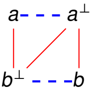
(-) Categorical and Algebraic Semantics

(-) Topological correctness criteria:

- Correctness criterion for GS



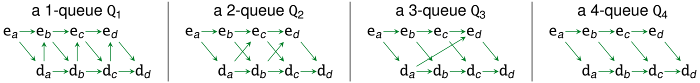
Satisfies Retoré's Criterion
Provable in GS



Satisfies Retoré's Criterion
NOT provable in GS

- Correctness criterion for BV $\Longrightarrow$ criteria for GV and GS

(-) Full homomorphism requires new tools



$$\vdash_{BV} Q_1 \multimap Q_2 \quad \vdash_{GV^{sl}} Q_2 \multimap Q_3 \quad \vdash_{???} Q_3 \multimap Q_4$$

(-) Applications to Games/Petri Nets/Event Structures/Concurrency/Verification

Thank you