The Composition of Combinatorial Flows

Giti Omidvar and Lutz Straßburger

Inria Saclay
Ecole Polytechnique

SYCO 10, Edinburgh, 20 December 2022

Preliminaries: Open Deduction

$$\frac{\mathsf{t}}{a \vee \bar{a}}\ \mathsf{ai}{\downarrow} \qquad \frac{(A \vee B) \wedge C}{A \vee (B \wedge C)}\ \mathsf{s} \qquad \frac{\bar{a} \wedge a}{\mathsf{f}}\ \mathsf{ai}{\uparrow}$$

$$\frac{a \vee a}{a}\ \mathsf{ac}{\downarrow} \qquad \frac{(A \wedge C) \vee (B \wedge D)}{(A \vee B) \wedge (C \vee D)}\ \mathsf{m} \qquad \frac{a}{a \wedge a}\ \mathsf{ac}{\uparrow}$$

$$\frac{\mathsf{f}}{a}\ \mathsf{aw}{\downarrow} \qquad \frac{\mathsf{f}}{\mathsf{t}}\ \mathsf{mix} \qquad \frac{a}{\mathsf{t}}\ \mathsf{aw}{\uparrow}$$

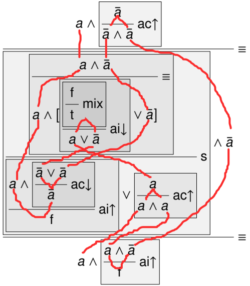
Preliminaries: Open Deduction

$$\dfrac{\dfrac{a \wedge \left(\dfrac{\bar a}{\bar a \wedge \bar a}\,\mathsf{ac}\!\uparrow\right)}{\left(\dfrac{\dfrac{a\wedge \bar a}{a \wedge \left[\dfrac{\left(\dfrac{\mathsf f}{\mathsf t}\,\mathsf{mix}\right)}{a \vee \bar a}\,\mathsf{ai}\!\downarrow \vee \bar a\right]}\,\equiv}{\left(\dfrac{a\wedge\left(\dfrac{\bar a\vee\bar a}{\bar a}\,\mathsf{ac}\!\downarrow\right)}{\mathsf f}\,\mathsf{ai}\!\uparrow\right)\vee\left(\dfrac{a}{a\wedge a}\,\mathsf{ac}\!\uparrow\right)}\,\mathsf{s}\right)\wedge\bar a}\,\equiv}{a \wedge \left(\dfrac{a\wedge\bar a}{\mathsf f}\,\mathsf{ai}\!\uparrow\right)}\,\equiv$$

$$\dfrac{\mathsf t}{a \vee \bar a}\,\mathsf{ai}\!\downarrow \qquad \dfrac{(A \vee B)\wedge C}{A \vee (B \wedge C)}\,\mathsf{s} \qquad \dfrac{\bar a \wedge a}{\mathsf f}\,\mathsf{ai}\!\uparrow$$

$$\dfrac{a \vee a}{a}\,\mathsf{ac}\!\downarrow \qquad \dfrac{(A \wedge C)\vee(B \wedge D)}{(A \vee B)\wedge(C \vee D)}\,\mathsf{m} \qquad \dfrac{a}{a \wedge a}\,\mathsf{ac}\!\uparrow$$

$$\dfrac{\mathsf f}{a}\,\mathsf{aw}\!\downarrow \qquad \dfrac{\mathsf f}{\mathsf t}\,\mathsf{mix} \qquad \dfrac{a}{\mathsf t}\,\mathsf{aw}\!\uparrow$$



$$\frac{\mathsf{t}}{a \vee \bar{a}}\,\mathsf{ai}{\downarrow} \qquad \frac{(A \vee B) \wedge C}{A \vee (B \wedge C)}\,\mathsf{s} \qquad \frac{\bar{a} \wedge a}{\mathsf{f}}\,\mathsf{ai}{\uparrow}$$

$$\frac{a \vee a}{a}\,\mathsf{ac}{\downarrow} \qquad \frac{(A \wedge C) \vee (B \wedge D)}{(A \vee B) \wedge (C \vee D)}\,\mathsf{m} \qquad \frac{a}{a \wedge a}\,\mathsf{ac}{\uparrow}$$

$$\frac{\mathsf{f}}{a}\,\mathsf{aw}{\downarrow} \qquad \frac{\mathsf{f}}{\mathsf{t}}\,\mathsf{mix} \qquad \frac{a}{\mathsf{t}}\,\mathsf{aw}{\uparrow}$$

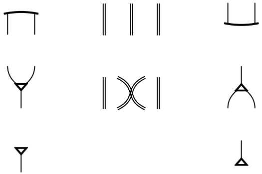
Preliminaries: Open Deduction and Atomic Flows

$$\dfrac{\dfrac{a \wedge \boxed{\dfrac{\bar a}{\bar a \wedge \bar a}\,\mathsf{ac}\!\uparrow}}{\boxed{\dfrac{\boxed{\dfrac{a \wedge \bar a}{a \wedge [\boxed{\dfrac{\boxed{\dfrac{\mathsf{f}}{\mathsf{t}}\,\mathsf{mix}}}{a \vee \bar a}\,\mathsf{ai}\!\downarrow} \vee \bar a]}\,\equiv}}{\boxed{\dfrac{a \wedge \boxed{\dfrac{\bar a \vee \bar a}{\bar a}\,\mathsf{ac}\!\downarrow}}{\mathsf{f}}\,\mathsf{ai}\!\uparrow} \vee \boxed{\dfrac{a}{a \wedge a}\,\mathsf{ac}\!\uparrow}}\,\mathsf{s}} \wedge \bar a}\,\equiv}{a \wedge \boxed{\dfrac{a \wedge \bar a}{\mathsf{f}}\,\mathsf{ai}\!\uparrow}}\,\equiv$$

$$\dfrac{\mathsf{t}}{a \vee \bar a}\,\mathsf{ai}\!\downarrow \qquad \dfrac{(A \vee B) \wedge C}{A \vee (B \wedge C)}\,\mathsf{s} \qquad \dfrac{\bar a \wedge a}{\mathsf{f}}\,\mathsf{ai}\!\uparrow$$

$$\dfrac{a \vee a}{a}\,\mathsf{ac}\!\downarrow \qquad \dfrac{(A \wedge C) \vee (B \wedge D)}{(A \vee B) \wedge (C \vee D)}\,\mathsf{m} \qquad \dfrac{a}{a \wedge a}\,\mathsf{ac}\!\uparrow$$

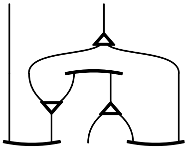
$$\dfrac{\mathsf{f}}{a}\,\mathsf{aw}\!\downarrow \qquad \dfrac{\mathsf{f}}{\mathsf{t}}\,\mathsf{mix} \qquad \dfrac{a}{\mathsf{t}}\,\mathsf{aw}\!\uparrow$$



Alessio Guglielmi and Tom Gundersen, LMCS 2008

Preliminaries: Open Deduction and Atomic Flows

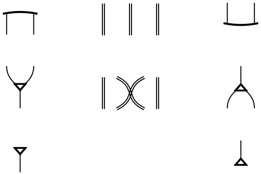
$$a \wedge \dfrac{\bar a}{\bar a \wedge \bar a}\,\mathsf{ac}\!\uparrow$$
$$\equiv$$
$$\left(\dfrac{\dfrac{a \wedge \bar a}{a \wedge \left[\dfrac{\dfrac{\mathsf{f}}{\mathsf{t}}\,\mathsf{mix}}{a \vee \bar a}\,\mathsf{ai}\!\downarrow \vee \bar a\right]}\,\equiv}{\dfrac{a \wedge \dfrac{\bar a \vee \bar a}{\bar a}\,\mathsf{ac}\!\downarrow}{\mathsf{f}}\,\mathsf{ai}\!\uparrow \;\vee\; \dfrac{a}{a \wedge a}\,\mathsf{ac}\!\uparrow}\,\mathsf{s}\right) \wedge \bar a$$
$$\equiv$$
$$a \wedge \dfrac{a \wedge \bar a}{\mathsf{f}}\,\mathsf{ai}\!\uparrow$$



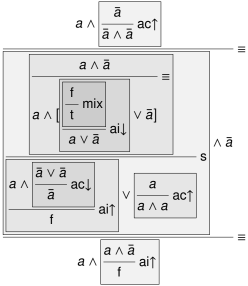
$$\dfrac{\mathsf{t}}{a \vee \bar a}\,\mathsf{ai}\!\downarrow \qquad \dfrac{(A \vee B) \wedge C}{A \vee (B \wedge C)}\,\mathsf{s} \qquad \dfrac{\bar a \wedge a}{\mathsf{f}}\,\mathsf{ai}\!\uparrow$$

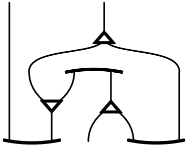
$$\dfrac{a \vee a}{a}\,\mathsf{ac}\!\downarrow \qquad \dfrac{(A \wedge C) \vee (B \wedge D)}{(A \vee B) \wedge (C \vee D)}\,\mathsf{m} \qquad \dfrac{a}{a \wedge a}\,\mathsf{ac}\!\uparrow$$

$$\dfrac{\mathsf{f}}{a}\,\mathsf{aw}\!\downarrow \qquad \dfrac{\mathsf{f}}{\mathsf{t}}\,\mathsf{mix} \qquad \dfrac{a}{\mathsf{t}}\,\mathsf{aw}\!\uparrow$$



Alessio Guglielmi and Tom Gundersen, LMCS 2008

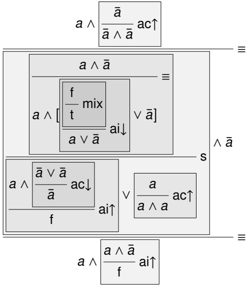


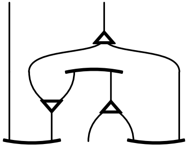


1. We cannot read back a proof from atomic flows

Anupam Das, RTA 2013

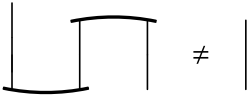
Alessio Guglielmi and Tom Gundersen, LMCS 2008





1. We cannot read back a proof from atomic flows

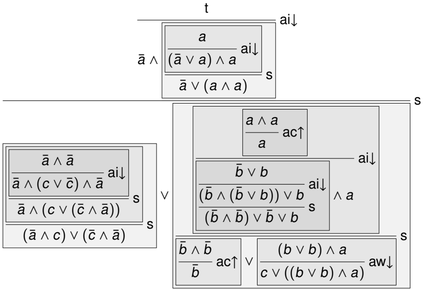
2. yanking is not possible

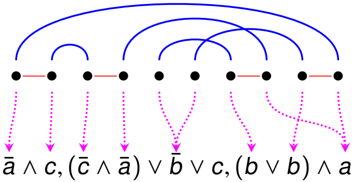


Anupam Das, RTA 2013

Alessio Guglielmi and Tom Gundersen, LMCS 2008

Preliminaries: Combinatorial Proofs



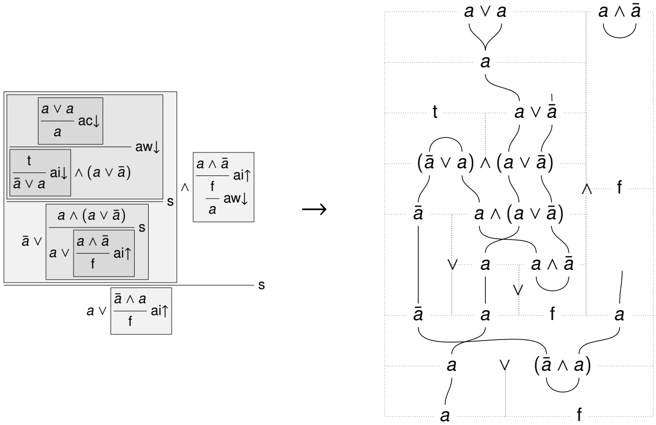


- Total seperation of linear part and resource management of the proof → size explosion

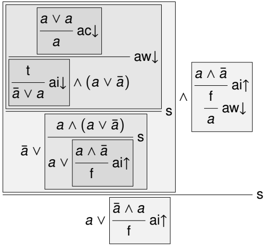
Dominic Hughes, Annals of Mathematics 2006

$$
\frac{
\boxed{
\frac{
\boxed{
\frac{
\boxed{\frac{a \vee a}{a}\,\mathsf{ac}{\downarrow}}
}{
\boxed{\frac{\mathsf{t}}{\bar{a} \vee a}\,\mathsf{ai}{\downarrow}} \wedge (a \vee \bar{a})
}\,\mathsf{aw}{\downarrow}
}
}{
\bar{a} \vee \boxed{\frac{a \wedge (a \vee \bar{a})}{a \vee \boxed{\frac{a \wedge \bar{a}}{\mathsf{f}}\,\mathsf{ai}{\uparrow}}}\,\mathsf{s}}
}\,\mathsf{s}
}
\wedge
\boxed{
\frac{\frac{a \wedge \bar{a}}{\mathsf{f}}\,\mathsf{ai}{\uparrow}}{a}\,\mathsf{aw}{\downarrow}
}
}{
a \vee \boxed{\frac{\bar{a} \wedge a}{\mathsf{f}}\,\mathsf{ai}{\uparrow}}
}\,\mathsf{s}
$$

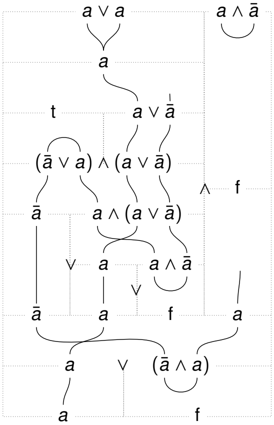
From Open Deduction to Preflows



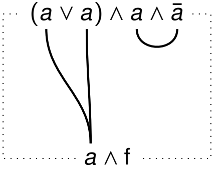
From Open Deduction to Preflows



→

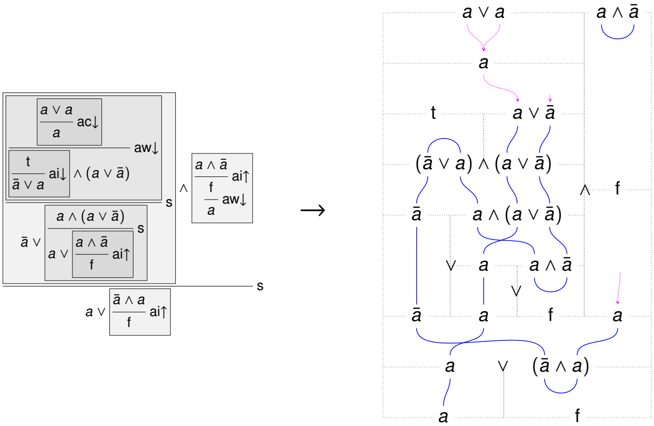


→



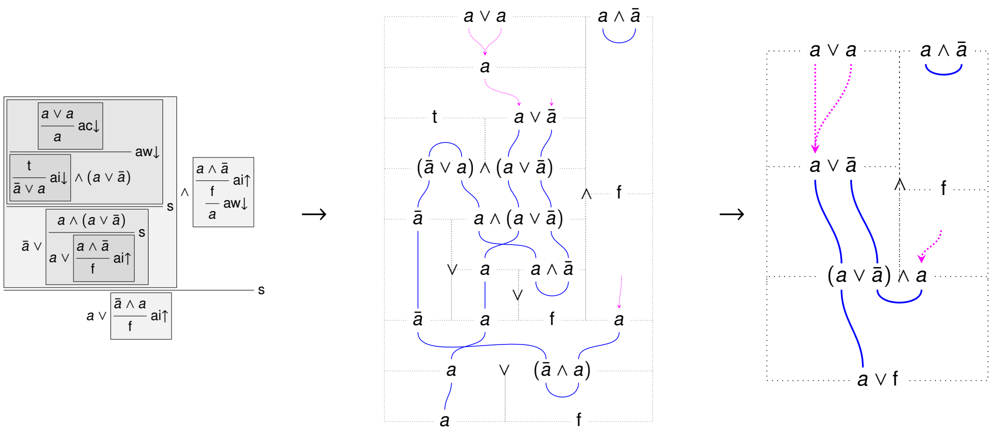
Corresponds to B-nets

B-nets: François Lamarche and Lutz Straßburger, TLCA 2005



a ∨ a
ac↓
a
aw↓
t
ai↓
ā ∨ a
∧ (a ∨ ā)
s
∧
a ∧ ā
ai↑
f
aw↓
a
a ∧ (a ∨ ā)
s
ā ∨
a ∨
a ∧ ā
ai↑
f
s
a ∨
ā ∧ a
ai↑
f
→
a ∨ a
a ∧ ā
a
t
a ∨ ā
(ā ∨ a) ∧ (a ∨ ā)
∧
f
ā
a ∧ (a ∨ ā)
∨
a
a ∧ ā
∨
ā
a
f
a
a
∨
(ā ∧ a)
a
f

From Open Deduction to Combinatorial Flows



Formulas:

$$A, B := \mathsf{t} \mid \mathsf{f} \mid a \mid \overline{a} \mid A \vee B \mid A \wedge B$$

Formulas:

$$A, B := \ \mathsf{t} \mid \mathsf{f} \mid a \mid \overline{a} \mid A \vee B \mid A \wedge B$$

$$A \wedge B \equiv B \wedge A \qquad (A \wedge B) \wedge C \equiv A \wedge (B \wedge C) \qquad A \wedge \mathsf{t} \equiv A \qquad \mathsf{t} \vee \mathsf{t} \equiv \mathsf{t}$$

$$A \vee B \equiv B \vee A \qquad (A \vee B) \vee C \equiv A \vee (B \vee C) \qquad A \vee \mathsf{f} \equiv A \qquad \mathsf{f} \wedge \mathsf{f} \equiv \mathsf{f}$$

Formulas:

$$A, B := \mathsf{t} \mid \mathsf{f} \mid a \mid \overline{a} \mid A \vee B \mid A \wedge B$$

$$A \wedge B \equiv B \wedge A \qquad (A \wedge B) \wedge C \equiv A \wedge (B \wedge C) \qquad A \wedge \mathsf{t} \equiv A \qquad \mathsf{t} \vee \mathsf{t} \equiv \mathsf{t}$$
$$A \vee B \equiv B \vee A \qquad (A \vee B) \vee C \equiv A \vee (B \vee C) \qquad A \vee \mathsf{f} \equiv A \qquad \mathsf{f} \wedge \mathsf{f} \equiv \mathsf{f}$$

Unit-free Formulas:

$$A, B := a \mid \overline{a} \mid A \vee B \mid A \wedge B$$

Formulas:

$$A, B := \mathrm{t} \mid \mathrm{f} \mid a \mid \overline{a} \mid A \vee B \mid A \wedge B$$

$$A \wedge B \equiv B \wedge A \qquad (A \wedge B) \wedge C \equiv A \wedge (B \wedge C) \qquad A \wedge \mathrm{t} \equiv A \qquad \mathrm{t} \vee \mathrm{t} \equiv \mathrm{t}$$
$$A \vee B \equiv B \vee A \qquad (A \vee B) \vee C \equiv A \vee (B \vee C) \qquad A \vee \mathrm{f} \equiv A \qquad \mathrm{f} \wedge \mathrm{f} \equiv \mathrm{f}$$

Unit-free Formulas:

$$A, B := a \mid \overline{a} \mid A \vee B \mid A \wedge B$$

Pure Formulas: $A \equiv \mathrm{t}$ or $A \equiv \mathrm{f}$ or A is equivalent to a unit-free formula.

Graph of a formula

Graph of a formula

- $\mathcal{G}(\mathsf{t}) = \mathcal{G}(\mathsf{f})$: empty graph

Graph of a formula

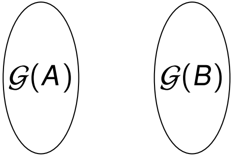
- $\mathcal{G}(\mathsf{t}) = \mathcal{G}(\mathsf{f})$: empty graph
- $\mathcal{G}(a)$: $\bullet_a$

Graph of a formula

- $\mathcal{G}(\mathsf{t}) = \mathcal{G}(\mathsf{f})$: empty graph
- $\mathcal{G}(a)$: $\bullet_a$
- $\mathcal{G}(\bar{a})$: $\bullet_{\bar{a}}$

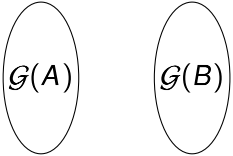
Graph of a formula

- $\mathcal{G}(\mathrm{t}) = \mathcal{G}(\mathrm{f})$: empty graph
- $\mathcal{G}(a)$: $\bullet_a$
- $\mathcal{G}(\bar{a})$: $\bullet_{\bar{a}}$
- $\mathcal{G}(A \vee B)$:

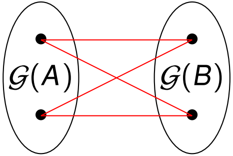


Graph of a formula

- $\mathcal{G}(\mathrm{t}) = \mathcal{G}(\mathrm{f})$: empty graph
- $\mathcal{G}(a)$: $\bullet_{a}$
- $\mathcal{G}(\bar{a})$: $\bullet_{\bar{a}}$
- $\mathcal{G}(A \vee B)$:

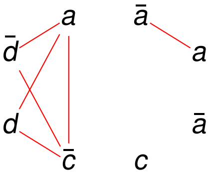


- $\mathcal{G}(A \wedge B)$:



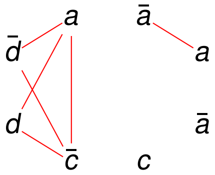
Graph of a formula

$$((d \vee \bar{d}) \wedge (\bar{c} \wedge a)) \vee ((c \vee \bar{a}) \vee (a \wedge \bar{a}))$$

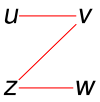


Graph of a formula

$((d \vee \bar{d}) \wedge (\bar{c} \wedge a)) \vee ((c \vee \bar{a}) \vee (a \wedge \bar{a}))$

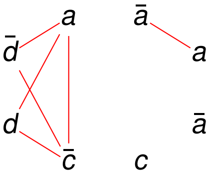


A **cograph** is a graph without $\mathcal{P}_4$:

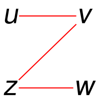


Graph of a formula

$((d \vee \bar{d}) \wedge (\bar{c} \wedge a)) \vee ((c \vee \bar{a}) \vee (a \wedge \bar{a}))$



A **cograph** is a graph without $\mathcal{P}_4$:



Theorem

A graph $\mathcal{G}$ is graph of a formula A if and only if $\mathcal{G}$ is a cograph.

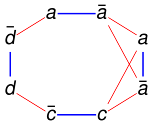
Multiplicative Flows

A triple $\phi = \langle A, B, \mathbb{B}_\phi \rangle$ is an **m-flow** if A and B are pure formulas, $\mathbb{B}_\phi$ is a perfect matching on the atom occurences of $\bar{A} \vee B$

Christian Retore, Theoretical Computer Science 2003 (Handsome Proof Nets)

A triple $\phi = \langle A, B, \mathbb{B}_\phi \rangle$ is an **m-flow** if A and B are pure formulas, $\mathbb{B}_\phi$ is a perfect matching on the atom occurences of $\bar{A} \vee B$ such that the underlying RB-cograph $\mathcal{G}(\phi)$ is æ-acyclic.

correctness criterion:

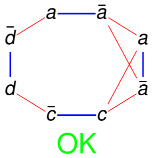


Christian Retore, Theoretical Computer Science 2003 (Handsome Proof Nets)

Multiplicative Flows

A triple $\phi = \langle A, B, \mathbb{B}_\phi \rangle$ is an **m-flow** if A and B are pure formulas, $\mathbb{B}_\phi$ is a perfect matching on the atom occurences of $\bar{A} \vee B$ such that the underlying RB-cograph $\mathcal{G}(\phi)$ is æ-acyclic.

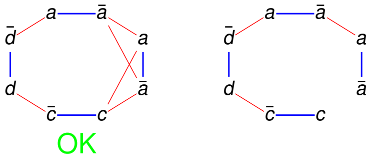
correctness criterion:



Christian Retore, Theoretical Computer Science 2003 (Handsome Proof Nets)

A triple $\phi = \langle A, B, \mathbb{B}_\phi \rangle$ is an **m-flow** if A and B are pure formulas, $\mathbb{B}_\phi$ is a perfect matching on the atom occurences of $\bar{A} \vee B$ such that the underlying RB-cograph $\mathcal{G}(\phi)$ is æ-acyclic.

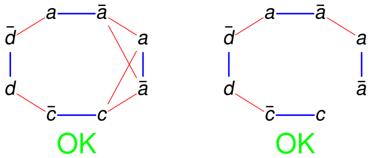
correctness criterion:



Christian Retore, Theoretical Computer Science 2003 (Handsome Proof Nets)

A triple $\phi = \langle A, B, \mathbb{B}_\phi \rangle$ is an **m-flow** if A and B are pure formulas, $\mathbb{B}_\phi$ is a perfect matching on the atom occurences of $\bar{A} \vee B$ such that the underlying RB-cograph $\mathcal{G}(\phi)$ is æ-acyclic.

correctness criterion:

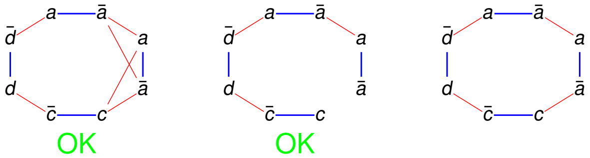


Christian Retore, Theoretical Computer Science 2003 (Handsome Proof Nets)

Multiplicative Flows

A triple $\phi = \langle A, B, \mathbb{B}_\phi \rangle$ is an **m-flow** if A and B are pure formulas, $\mathbb{B}_\phi$ is a perfect matching on the atom occurences of $\bar{A} \vee B$ such that the underlying RB-cograph $\mathcal{G}(\phi)$ is æ-acyclic.

correctness criterion:

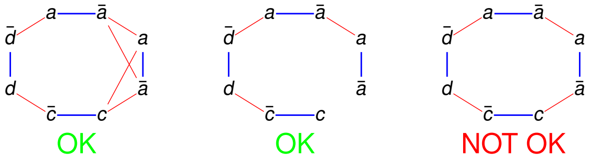


Christian Retore, Theoretical Computer Science 2003 (Handsome Proof Nets)

Multiplicative Flows

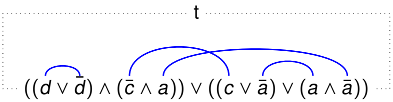
A triple $\phi = \langle A, B, \mathbb{B}_\phi \rangle$ is an **m-flow** if A and B are pure formulas, $\mathbb{B}_\phi$ is a perfect matching on the atom occurences of $\bar{A} \vee B$ such that the underlying RB-cograph $\mathcal{G}(\phi)$ is æ-acyclic.

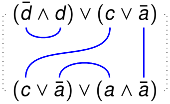
correctness criterion:

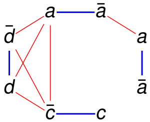


Christian Retore, Theoretical Computer Science 2003 (Handsome Proof Nets)

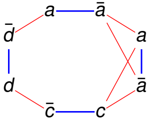
A triple $\phi = \langle A, B, \mathbb{B}_\phi \rangle$ is an **m-flow** if A and B are pure formulas, $\mathbb{B}_\phi$ is a perfect matching on the atom occurences of $\bar{A} \vee B$ such that the underlying RB-cograph $\mathcal{G}(\phi)$ is æ-acyclic.



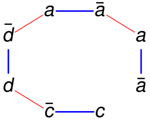




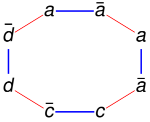
correctness criterion:



OK



OK



NOT OK

Christian Retore, Theoretical Computer Science 2003 (Handsome Proof Nets)

Multiplicative Flows

Theorem

Let $\begin{array}{c} A \\ \mathcal{D} \Vert \{\mathsf{ai}\downarrow, \mathsf{ai}\uparrow, \mathsf{s}, \mathsf{mix}\} \\ B \end{array}$ be a derivation. If A and B are pure, then the translation of $\mathcal{D}$ is an m-flow.

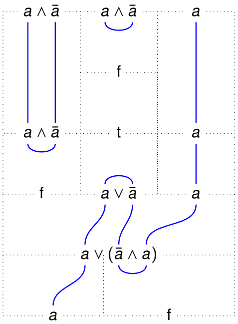
Theorem

Let $\phi = \langle A, B, \mathbb{B}_\phi \rangle$ be an m-flow. Then there is a derivation $\begin{array}{c} A \\ \mathcal{D} \Vert \{\mathsf{ai}\downarrow, \mathsf{ai}\uparrow, \mathsf{s}, \mathsf{mix}\} \\ B \end{array}$ whose translation is ϕ.

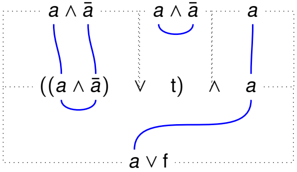
$$\frac{\mathsf{t}}{a \vee \bar{a}}\,\mathsf{ai}\downarrow \qquad \frac{(A \vee B) \wedge C}{A \vee (B \wedge C)}\,\mathsf{s} \qquad \frac{\bar{a} \wedge a}{\mathsf{f}}\,\mathsf{ai}\uparrow \qquad \frac{\mathsf{f}}{\mathsf{t}}\,\mathsf{mix}$$

$$
\dfrac{\dfrac{[(a \wedge \bar{a}) \vee \boxed{\dfrac{\dfrac{a \wedge \bar{a}}{\mathsf{f}}\,\mathsf{ai}\!\uparrow}{\mathsf{t}}\,\mathsf{mix}}] \wedge a}{[\boxed{\dfrac{a \wedge \bar{a}}{\mathsf{f}}\,\mathsf{ai}\!\uparrow} \vee \boxed{\dfrac{\mathsf{t}}{a \vee a}\,\mathsf{ai}\!\downarrow}] \wedge a}\,\equiv}{a \vee \boxed{\dfrac{a \wedge \bar{a}}{\mathsf{f}}\,\mathsf{ai}\!\uparrow}}\,\mathsf{s}
$$

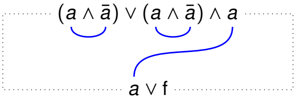
$\rightarrow$



$\downarrow$



$\leftarrow$



Additive Flows

- A triple $\phi = \langle A, B, f^{\downarrow}_{\phi} \rangle$ is an **a$^{\downarrow}$-flow** if A and B are pure, and $A \neq \mathrm{t}$, and $f^{\downarrow}_{\phi}$ is a skew fibration $f^{\downarrow}_{\phi} \colon \mathcal{G}(A) \to \mathcal{G}(B)$.

Additive Flows

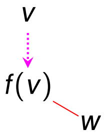
- A triple $\phi = \langle A, B, f^{\downarrow}_{\phi} \rangle$ is an **a$^{\downarrow}$-flow** if A and B are pure, and $A \neq \mathsf{t}$, and $f^{\downarrow}_{\phi}$ is a skew fibration $f^{\downarrow}_{\phi} \colon \mathcal{G}(A) \to \mathcal{G}(B)$.

A **skew fibration** is a graph homomorphism $f \colon \mathcal{G} \to \mathcal{H}$ such that

Additive Flows

- A triple $\phi = \langle A, B, f_\phi^\downarrow \rangle$ is an **a$^\downarrow$-flow** if A and B are pure, and $A \neq$ t, and $f_\phi^\downarrow$ is a skew fibration $f_\phi^\downarrow \colon \mathcal{G}(A) \to \mathcal{G}(B)$.

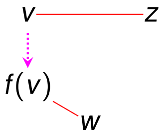
A **skew fibration** is a graph homomorphism $f \colon \mathcal{G} \to \mathcal{H}$ such that for every $v \in V_G$ and $w \in V_H$, with $f(v)w \in \mathcal{E}_H$,



Additive Flows

- A triple $\phi = \langle A, B, f_\phi^\downarrow \rangle$ is an **a**$^\downarrow$**-flow** if A and B are pure, and $A \neq$ t, and $f_\phi^\downarrow$ is a skew fibration $f_\phi^\downarrow : \mathcal{G}(A) \to \mathcal{G}(B)$.

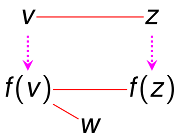
A **skew fibration** is a graph homomorphism $f : \mathcal{G} \to \mathcal{H}$ such that for every $v \in V_G$ and $w \in V_H$, with $f(v)w \in \mathcal{E}_H$, there exists $z \in \mathcal{G}$ with the edge $vz \in \mathcal{E}_G$



Additive Flows

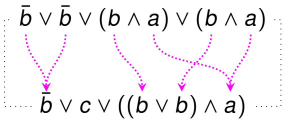
- A triple $\phi = \langle A, B, f_\phi^\downarrow \rangle$ is an **a$^\downarrow$-flow** if A and B are pure, and $A \neq$ t, and $f_\phi^\downarrow$ is a skew fibration $f_\phi^\downarrow \colon \mathcal{G}(A) \to \mathcal{G}(B)$.

A **skew fibration** is a graph homomorphism $f \colon \mathcal{G} \to \mathcal{H}$ such that for every $v \in V_G$ and $w \in V_H$, with $f(v)w \in \mathcal{E}_H$, there exists $z \in \mathcal{G}$ with the edge $vz \in \mathcal{E}_G$ such that the edge $f(z)w$ does not exist in $\mathcal{H}$.



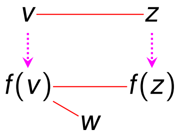
Additive Flows

- A triple $\phi = \langle A, B, f_\phi^\downarrow \rangle$ is an **a$^\downarrow$-flow** if A and B are pure, and $A \neq \mathsf{t}$, and $f_\phi^\downarrow$ is a skew fibration $f_\phi^\downarrow : \mathcal{G}(A) \to \mathcal{G}(B)$.

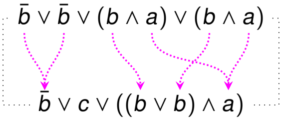


a$^\downarrow$-flow

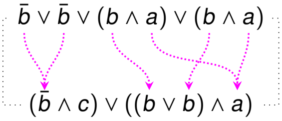
A **skew fibration** is a graph homomorphism $f : \mathcal{G} \to \mathcal{H}$ such that for every $v \in V_G$ and $w \in V_H$, with $f(v)w \in \mathcal{E}_H$, there exists $z \in \mathcal{G}$ with the edge $vz \in \mathcal{E}_G$ such that the edge $f(z)w$ does not exist in $\mathcal{H}$.



- A triple $\phi = \langle A, B, f_\phi^\downarrow \rangle$ is an **a$^\downarrow$-flow** if A and B are pure, and $A \neq$ t, and $f_\phi^\downarrow$ is a skew fibration $f_\phi^\downarrow \colon \mathcal{G}(A) \to \mathcal{G}(B)$.

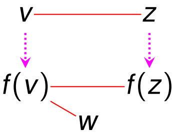


a$^\downarrow$-flow



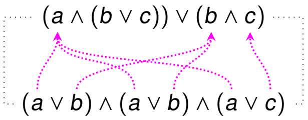
not a skew fibration

A **skew fibration** is a graph homomorphism $f \colon \mathcal{G} \to \mathcal{H}$ such that for every $v \in V_G$ and $w \in V_H$, with $f(v)w \in \mathcal{E}_H$, there exists $z \in \mathcal{G}$ with the edge $vz \in \mathcal{E}_G$ such that the edge $f(z)w$ does not exist in $\mathcal{H}$.

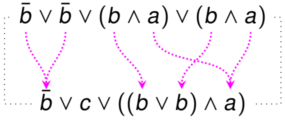


Additive Flows

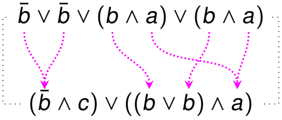
- A triple $\phi = \langle A, B, f^{\downarrow}_{\phi} \rangle$ is an **a$^{\downarrow}$-flow** if A and B are pure, and $A \neq$ t, and $f^{\downarrow}_{\phi}$ is a skew fibration $f^{\downarrow}_{\phi} \colon \mathcal{G}(A) \to \mathcal{G}(B)$.
- A triple $\phi = \langle C, D, f^{\uparrow}_{\phi} \rangle$ is an **a$^{\uparrow}$-flow** if C and D are pure, and $D \neq$ f, and $f^{\uparrow}_{\phi}$ is a skew fibration $f^{\uparrow}_{\phi} \colon \mathcal{G}(\overline{D}) \to \mathcal{G}(\overline{C})$.



a$^{\uparrow}$-flow

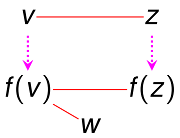


a$^{\downarrow}$-flow



not a skew fibration

A **skew fibration** is a graph homomorphism $f \colon \mathcal{G} \to \mathcal{H}$ such that for every $v \in V_G$ and $w \in V_H$, with $f(v)w \in \mathcal{E}_H$, there exists $z \in \mathcal{G}$ with the edge $vz \in \mathcal{E}_G$ such that the edge $f(z)w$ does not exist in $\mathcal{H}$.



Theorem

Let $\begin{array}{c} A \\ \mathcal{D}\Vert\{\mathsf{aw}{\downarrow},\mathsf{ac}{\downarrow},\mathsf{m}\} \\ B \end{array}$ be a derivation. If A and B are pure, then translation of $\mathcal{D}$ is an $\mathsf{a}^{\downarrow}$-flow. Dually, if A and B are pure in $\begin{array}{c} A \\ \mathcal{D}\Vert\{\mathsf{aw}{\uparrow},\mathsf{ac}{\uparrow},\mathsf{m}\} \\ B \end{array}$ then translation of $\mathcal{D}$ is an $\mathsf{a}^{\uparrow}$-flow.

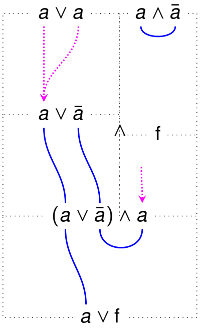
Theorem

Let $\phi = \langle A, B, f^{\downarrow}_{\phi} \rangle$ be an $\mathsf{a}^{\downarrow}$-flow. Then there is a derivation $\begin{array}{c} A \\ \mathcal{D}\Vert\{\mathsf{aw}{\downarrow},\mathsf{ac}{\downarrow},\mathsf{m}\} \\ B \end{array}$ whose translation is ϕ. For every $\mathsf{a}^{\uparrow}$-flow ψ we have $\begin{array}{c} A \\ \mathcal{D}\Vert\{\mathsf{aw}{\uparrow},\mathsf{ac}{\uparrow},\mathsf{m}\} \\ B \end{array}$ whose translation is ψ.

$$\frac{a \vee a}{a}\,\mathsf{ac}{\downarrow} \qquad \frac{\mathsf{f}}{a}\,\mathsf{aw}{\downarrow} \qquad \frac{(A \wedge C) \vee (B \wedge D)}{(A \vee B) \wedge (C \vee D)}\,\mathsf{m} \qquad \frac{a}{\mathsf{t}}\,\mathsf{aw}{\uparrow} \qquad \frac{a}{a \wedge a}\,\mathsf{ac}{\uparrow}$$

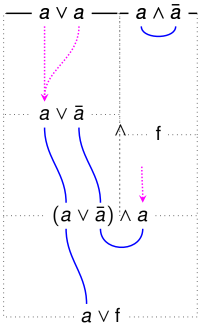
Pure Formulas: $A \equiv \mathrm{t}$ or $A \equiv \mathrm{f}$ or A is equivalent to a unit-free formula.

Slice of a Combinatorial flow:



Pure Formulas: $A \equiv \mathsf{t}$ or $A \equiv \mathsf{f}$ or A is equivalent to a unit-free formula.

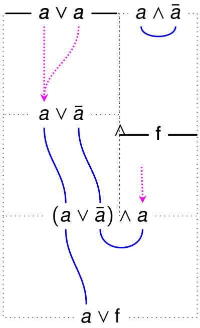
Slice of a Combinatorial flow:



$(a \vee a) \wedge (a \wedge \bar{a})$ pure

Pure Formulas: $A \equiv \mathsf{t}$ or $A \equiv \mathsf{f}$ or A is equivalent to a unit-free formula.

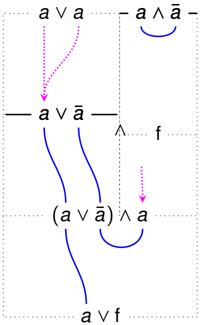
Slice of a Combinatorial flow:



$(a \vee a) \wedge \mathsf{f}$ not pure

Pure Formulas: $A \equiv \mathsf{t}$ or $A \equiv \mathsf{f}$ or A is equivalent to a unit-free formula.

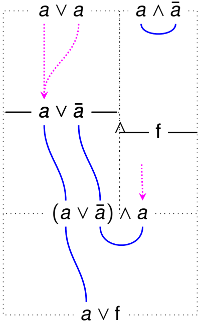
Slice of a Combinatorial flow:



$(a \vee \bar{a}) \wedge (a \wedge \bar{a})$ pure

Pure Formulas: $A \equiv \mathsf{t}$ or $A \equiv \mathsf{f}$ or A is equivalent to a unit-free formula.

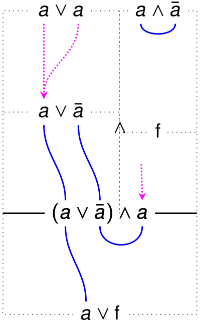
Slice of a Combinatorial flow:



$(a \vee \bar{a}) \wedge \mathsf{f}$ not pure

Pure Formulas: $A \equiv \mathsf{t}$ or $A \equiv \mathsf{f}$ or A is equivalent to a unit-free formula.

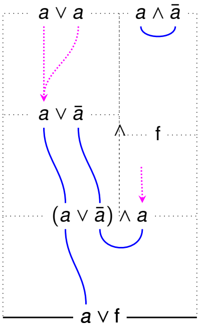
Slice of a Combinatorial flow:



$(a \vee \bar{a}) \wedge a$ pure

Pure Formulas: $A \equiv \mathsf{t}$ or $A \equiv \mathsf{f}$ or A is equivalent to a unit-free formula.

Slice of a Combinatorial flow:

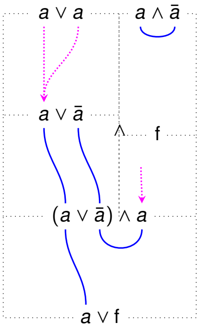


$a \vee \mathsf{f}$ pure

Purification of a formula:

$$A \wedge \mathsf{t} \rightsquigarrow A \qquad \mathsf{t} \wedge A \rightsquigarrow A \qquad A \vee \mathsf{t} \rightsquigarrow \mathsf{t} \qquad \mathsf{t} \vee A \rightsquigarrow \mathsf{t}$$
$$A \vee \mathsf{f} \rightsquigarrow A \qquad \mathsf{f} \vee A \rightsquigarrow A \qquad A \wedge \mathsf{f} \rightsquigarrow \mathsf{f} \qquad \mathsf{f} \wedge A \rightsquigarrow \mathsf{f}$$

Purification of combinatorial flows:



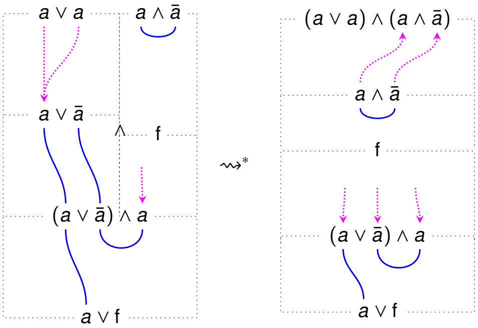
purification

Purification of a formula:

$$A \wedge \mathsf{t} \rightsquigarrow A \qquad \mathsf{t} \wedge A \rightsquigarrow A \qquad A \vee \mathsf{t} \rightsquigarrow \mathsf{t} \qquad \mathsf{t} \vee A \rightsquigarrow \mathsf{t}$$

$$A \vee \mathsf{f} \rightsquigarrow A \qquad \mathsf{f} \vee A \rightsquigarrow A \qquad A \wedge \mathsf{f} \rightsquigarrow \mathsf{f} \qquad \mathsf{f} \wedge A \rightsquigarrow \mathsf{f}$$

Purification of combinatorial flows:

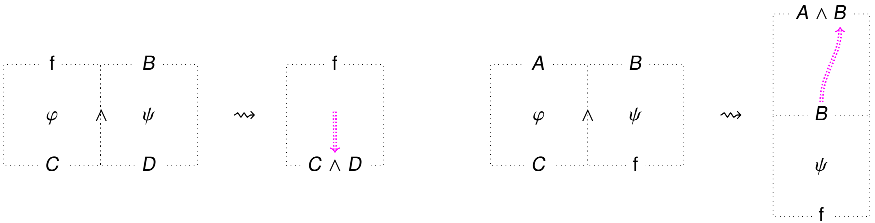


Purification of a formula:

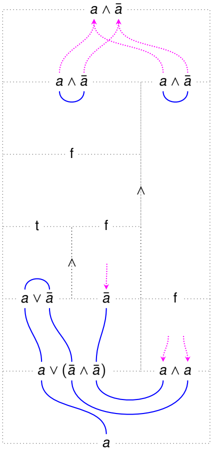
$$A \wedge \mathsf{t} \rightsquigarrow A \qquad \mathsf{t} \wedge A \rightsquigarrow A \qquad A \vee \mathsf{t} \rightsquigarrow \mathsf{t} \qquad \mathsf{t} \vee A \rightsquigarrow \mathsf{t}$$

$$A \vee \mathsf{f} \rightsquigarrow A \qquad \mathsf{f} \vee A \rightsquigarrow A \qquad A \wedge \mathsf{f} \rightsquigarrow \mathsf{f} \qquad \mathsf{f} \wedge A \rightsquigarrow \mathsf{f}$$

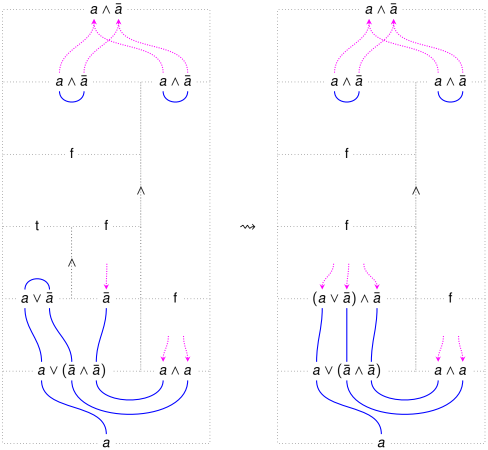
Purification of combinatorial flows:



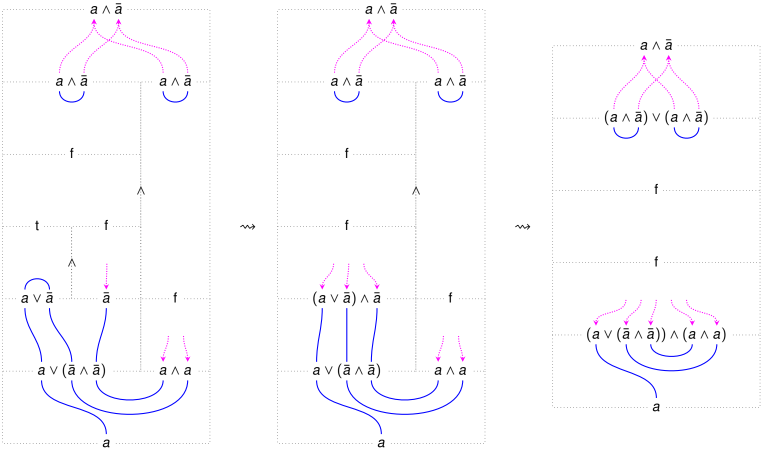
Purification Example



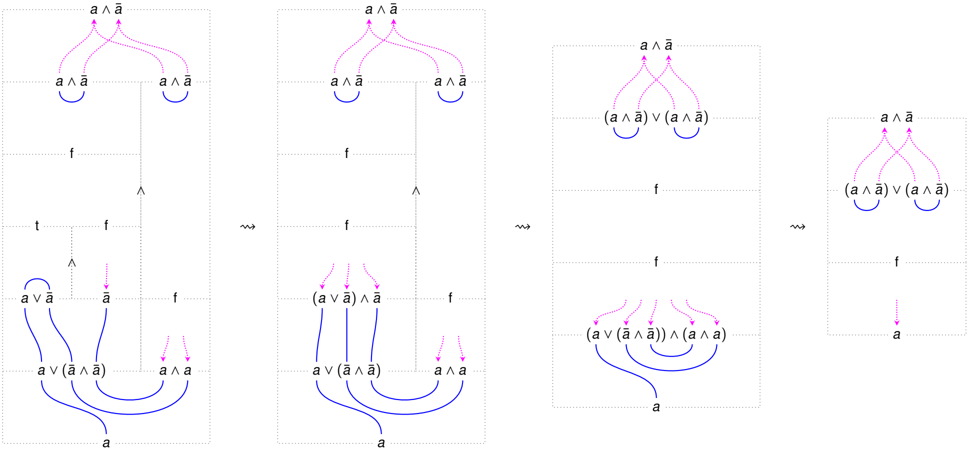
Purification Example



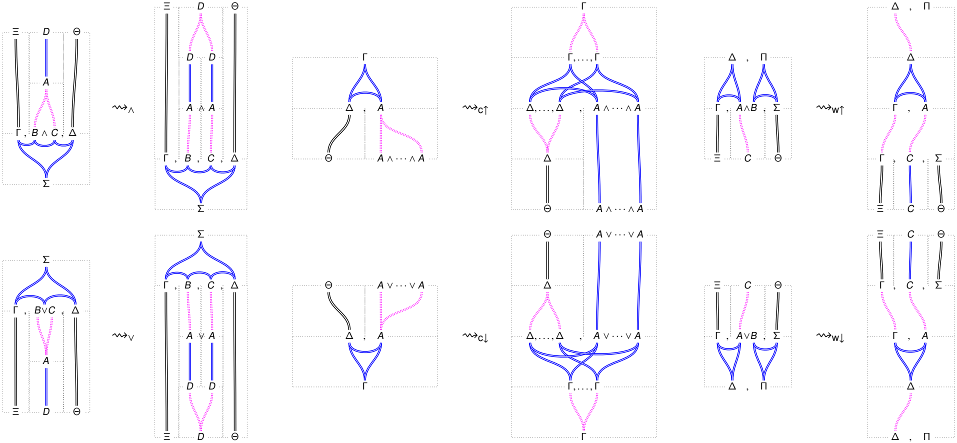
Purification Example



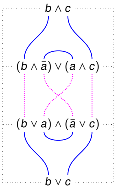
Purification Example



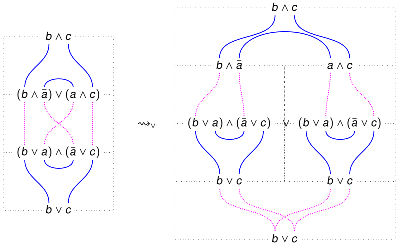
Normalization (Work in Progress)



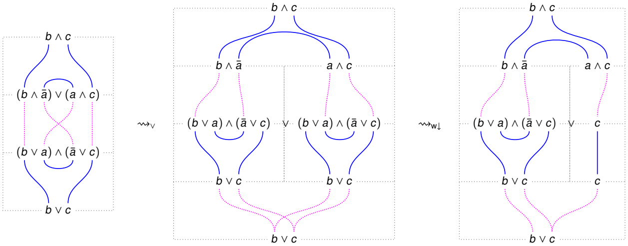
Normalization Example



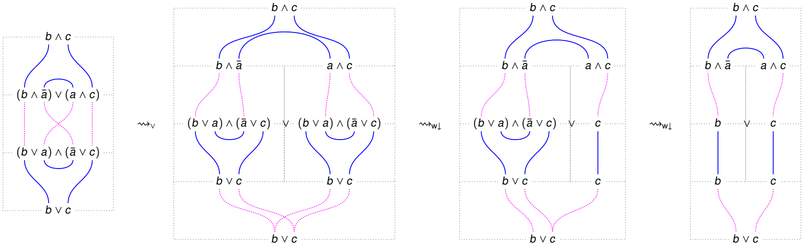
Normalization Example



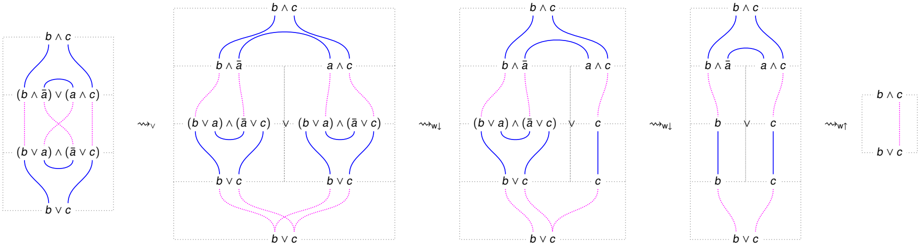
Normalization Example



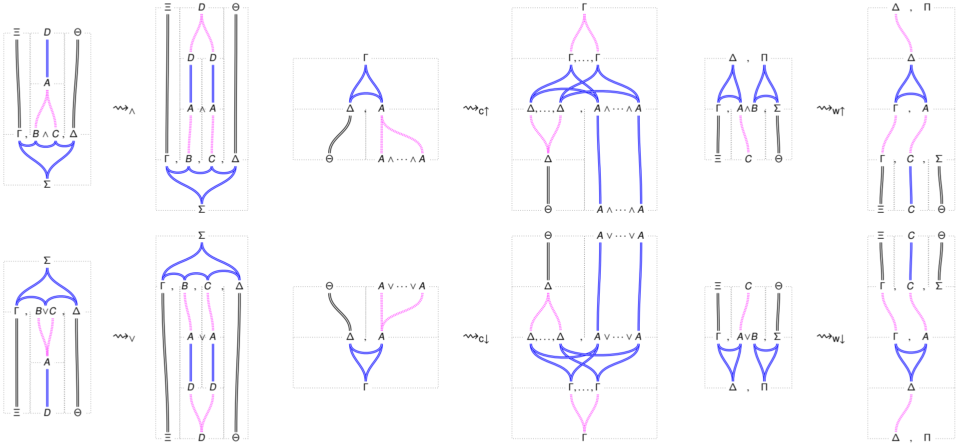
Normalization Example



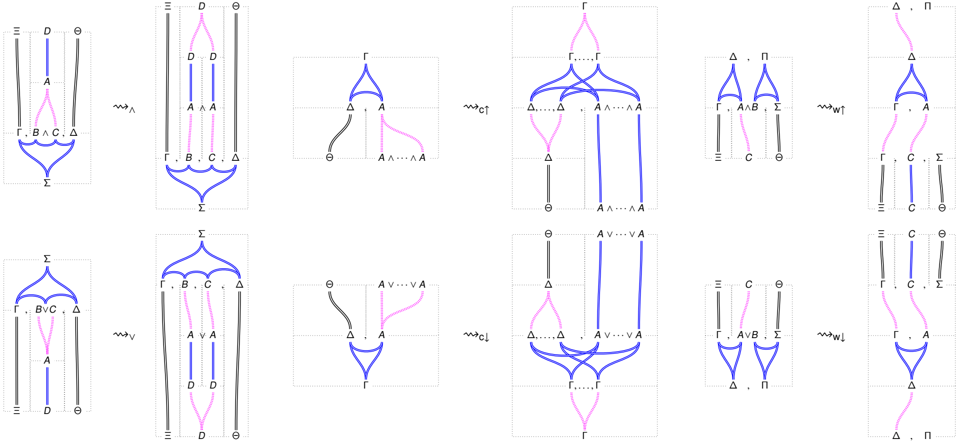
Normalization Example



Normalization (Work in Progress)

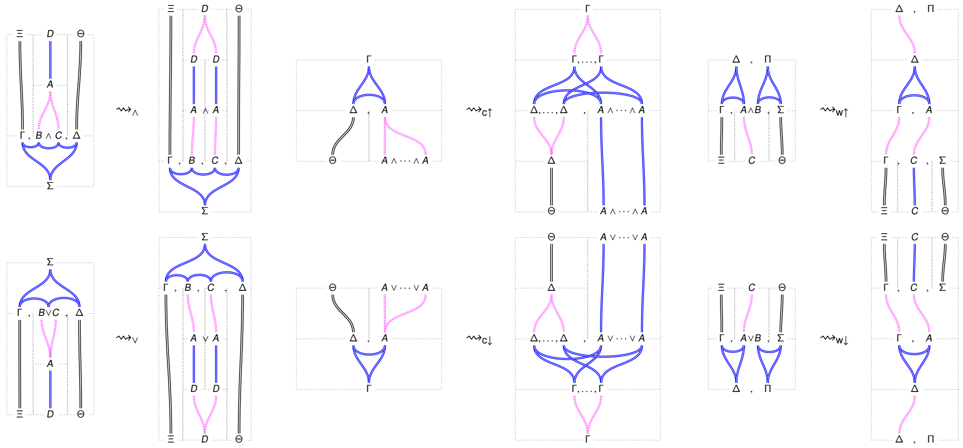


Normalization (Work in Progress)



NOT confluent

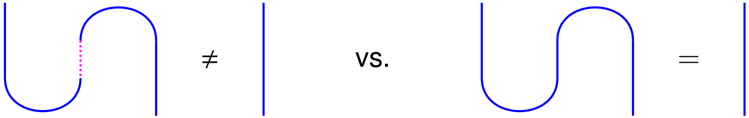
Normalization (Work in Progress)



NOT confluent and NOT terminating

What to remember from this talk?

What to remember from this talk?

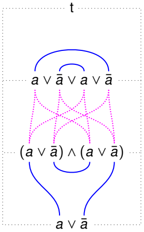


Future Work

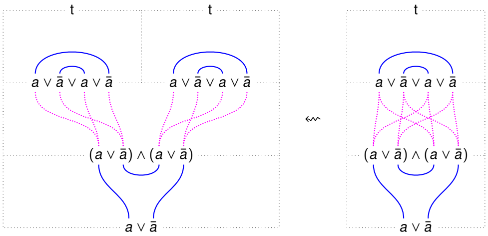
- Normalization Termination
- Proof identity
- Other Logics (Forexample: Modal Logic and Intuitionistic Logic)

What to remember from this talk?

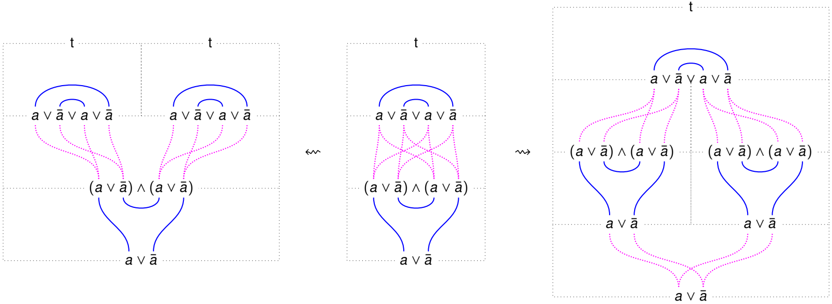
Normalization is not Confluent and not Terminating



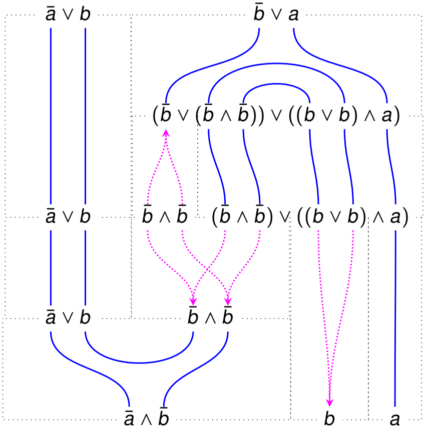
Normalization is not Confluent and not Terminating



Normalization is not Confluent and not Terminating



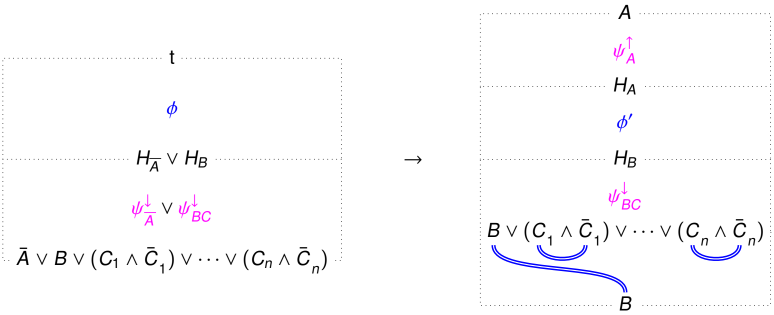
Yanking example5



A combinatorial proof with cuts for the sequent Γ is a combinatorial proof for the sequent $\Gamma, C_1 \wedge \bar{C}_1, \ldots, C_n \wedge \bar{C}_n$ where $C_1, \ldots C_n$ are cut formulas. (Everything is unit-free)

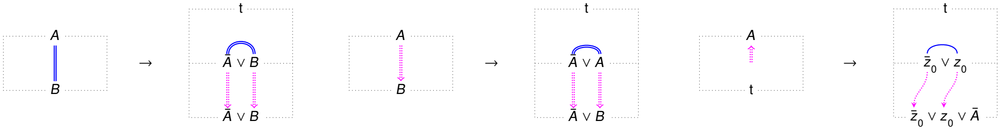
A combinatorial proof with cuts for the sequent Γ is a combinatorial proof for the sequent $\Gamma, C_1 \wedge \bar{C}_1, \ldots, C_n \wedge \bar{C}_n$ where $C_1, \ldots C_n$ are cut formulas. (Everything is unit-free)

Translating a combinatorial proof with cuts of $\bar{A}, B$ to a combinatorial flow from A to B:



A combinatorial proof with cuts for the sequent Γ is a combinatorial proof for the sequent $\Gamma, C_1 \wedge \bar{C}_1, \ldots, C_n \wedge \bar{C}_n$ where $C_1, \ldots C_n$ are cut formulas. (Everything is unit-free)

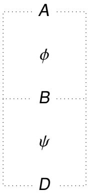
Translating a combinatorial flow with premise A and conclusion B to a combinatorial proof with cuts for $\bar{A}, B$:



Combinatorial Flows vs. Combinatorial Proofs with cuts

A combinatorial proof with cuts for the sequent Γ is a combinatorial proof for the sequent $\Gamma, C_1 \wedge \bar{C}_1, \ldots, C_n \wedge \bar{C}_n$ where $C_1, \ldots C_n$ are cut formulas. (Everything is unit-free)

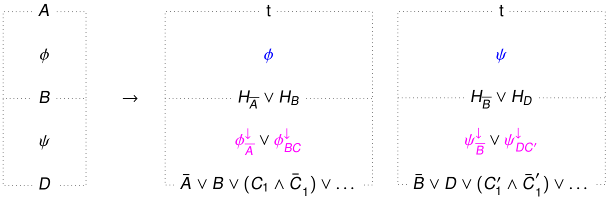
Translating a combinatorial flow with premise A and conclusion B to a combinatorial proof with cuts for $\bar{A}, B$:



Combinatorial Flows vs. Combinatorial Proofs with cuts

A combinatorial proof with cuts for the sequent Γ is a combinatorial proof for the sequent $\Gamma, C_1 \wedge \bar{C}_1, \ldots, C_n \wedge \bar{C}_n$ where $C_1, \ldots C_n$ are cut formulas. (Everything is unit-free)

Translating a combinatorial flow with premise A and conclusion B to a combinatorial proof with cuts for $\bar{A}, B$:



A combinatorial proof with cuts for the sequent Γ is a combinatorial proof for the sequent $\Gamma, C_1 \wedge \bar{C}_1, \ldots, C_n \wedge \bar{C}_n$ where $C_1, \ldots C_n$ are cut formulas. (Everything is unit-free)

Translating a combinatorial flow with premise A and conclusion B to a combinatorial proof with cuts for $\bar{A}, B$: