Sequentialization is as fun as bungee jumping

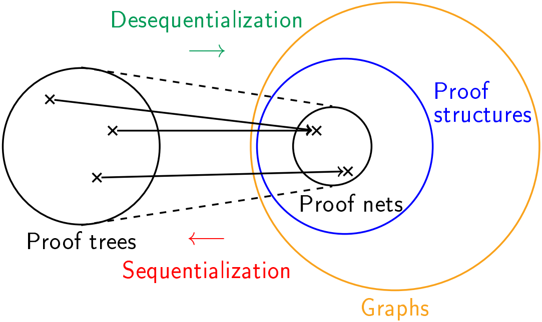
Rémi Di Guardia* Olivier Laurent*
Lorenzo Tortora de Falco† Lionel Vaux Auclair‡

*Lyon, †Rome, ‡Marseille

TLLA 2023, 1 July 2023

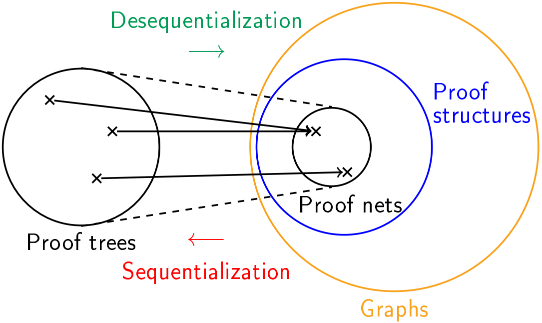
Introduction

Proof nets: graphical, more canonical representation of LL proofs



Introduction

Proof nets: graphical, more canonical representation of LL proofs

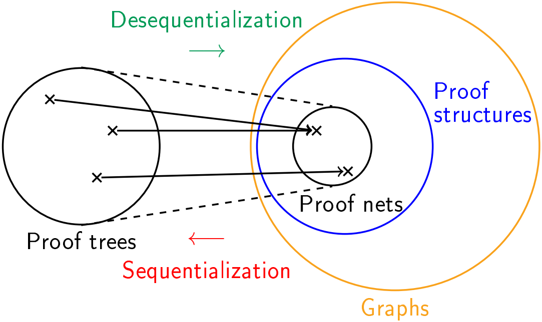


In (unit-free) MLL:
multiple **correctness criteria**,
proofs of sequentialization

Still sequentialization is not considered easy.

Introduction

Proof nets: graphical, more canonical representation of LL proofs



In (unit-free) MLL:
multiple **correctness criteria**,
proofs of sequentialization

Still sequentialization is not considered easy.

Objective of this talk: present a new simple proof of sequentialization

Outline

Unit-free Multiplicative Linear Logic with Mix

Formulas

$$A ::= X \mid X^{\perp} \mid A \otimes A \mid A \parr A$$

Orthogonality

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

Rules

$$\frac{}{\vdash A^{\perp}, A}\ (ax) \qquad \frac{\vdash A, \Gamma \quad \vdash B, \Delta}{\vdash A \otimes B, \Gamma, \Delta}\ (\otimes) \qquad \frac{\vdash A, B, \Gamma}{\vdash A \parr B, \Gamma}\ (\parr)$$

$$\frac{}{\vdash}\ (mix_0) \qquad \frac{\vdash \Gamma \quad \vdash \Delta}{\vdash \Gamma, \Delta}\ (mix_2)$$

Unit-free Multiplicative Linear Logic with Mix

Formulas

$$A ::= X \mid X^{\perp} \mid A \otimes A \mid A \parr A$$

Orthogonality

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

Rules

$$\frac{}{\vdash A^{\perp}, A}\,(ax) \qquad \frac{\vdash A, \Gamma \quad \vdash B, \Delta}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes) \qquad \frac{\vdash A, B, \Gamma}{\vdash A \parr B, \Gamma}\,(\parr)$$

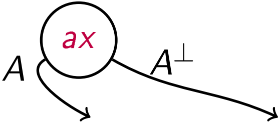
$$\frac{}{\vdash}\,(mix_0) \qquad \frac{\vdash \Gamma \quad \vdash \Delta}{\vdash \Gamma, \Delta}\,(mix_2)$$

Example of proof structure by desequentialization

$$
\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \qquad \dfrac{}{\vdash B, B^\perp}\,(ax)}{\vdash A \otimes B, A^\perp, B^\perp}\,(\otimes) \qquad \dfrac{}{\vdash C, C^\perp}\,(ax)}{\dfrac{\vdash A \otimes B, A^\perp, B^\perp, C, C^\perp}{\vdash A \otimes B, A^\perp \parr B^\perp, C, C^\perp}\,(\parr)}\,(mix_2)}{\vdash A \otimes B, (A^\perp \parr B^\perp) \parr C, C^\perp}\,(\parr)
$$

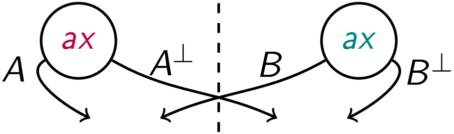
Example of proof structure by desequentialization

$$
\dfrac{\dfrac{\dfrac{\dfrac{\overline{\vdash A^{\perp}, A}\ (ax) \quad \overline{\vdash B, B^{\perp}}\ (ax)}{\vdash A \otimes B, A^{\perp}, B^{\perp}}\ (\otimes) \quad \overline{\vdash C, C^{\perp}}\ (ax)}{\vdash A \otimes B, A^{\perp}, B^{\perp}, C, C^{\perp}}\ (mix_2)}{\vdash A \otimes B, A^{\perp} \parr B^{\perp}, C, C^{\perp}}\ (\parr)}{\vdash A \otimes B, (A^{\perp} \parr B^{\perp}) \parr C, C^{\perp}}\ (\parr)
$$



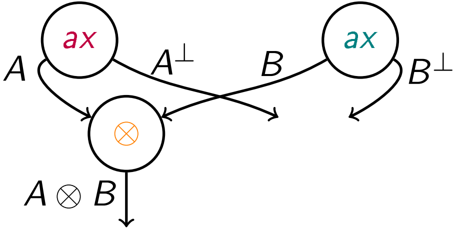
Example of proof structure by desequentialization

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash A^{\perp}, A}\,(ax) \qquad \dfrac{}{\vdash B, B^{\perp}}\,(ax)}{\vdash A \otimes B, A^{\perp}, B^{\perp}}\,(\otimes) \qquad \dfrac{}{\vdash C, C^{\perp}}\,(ax)}{\vdash A \otimes B, A^{\perp}, B^{\perp}, C, C^{\perp}}\,(mix_2)}{\vdash A \otimes B, A^{\perp} \parr B^{\perp}, C, C^{\perp}}\,(\parr)}{\vdash A \otimes B, (A^{\perp} \parr B^{\perp}) \parr C, C^{\perp}}\,(\parr)$$



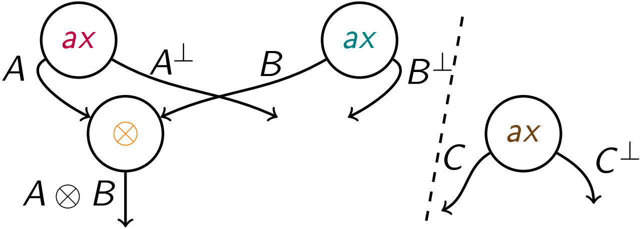
Example of proof structure by desequentialization

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \quad \dfrac{}{\vdash B, B^\perp}\,(ax)}{\vdash A \otimes B, A^\perp, B^\perp}\,(\otimes) \quad \dfrac{}{\vdash C, C^\perp}\,(ax)}{\vdash A \otimes B, A^\perp, B^\perp, C, C^\perp}\,(mix_2)}{\vdash A \otimes B, A^\perp \parr B^\perp, C, C^\perp}\,(\parr)}{\vdash A \otimes B, (A^\perp \parr B^\perp) \parr C, C^\perp}\,(\parr)
$$



Example of proof structure by desequentialization

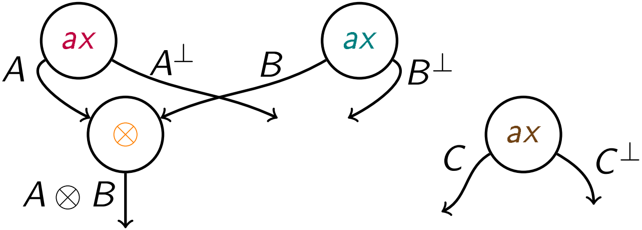
$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}(ax) \qquad \dfrac{}{\vdash B, B^\perp}(ax)}{\vdash A \otimes B, A^\perp, B^\perp}(\otimes) \qquad \dfrac{}{\vdash C, C^\perp}(ax)}{\vdash A \otimes B, A^\perp, B^\perp, C, C^\perp}(mix_2)}{\vdash A \otimes B, A^\perp ⅋ B^\perp, C, C^\perp}(⅋)}{\vdash A \otimes B, (A^\perp ⅋ B^\perp) ⅋ C, C^\perp}(⅋)}{}$$



Example of proof structure by desequentialization

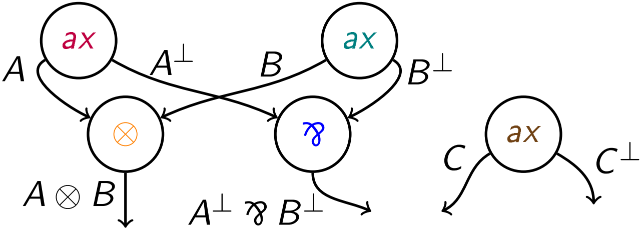
$$\dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \quad \dfrac{}{\vdash B, B^\perp}\,(ax)}{\vdash A \otimes B, A^\perp, B^\perp}\,(\otimes) \quad \dfrac{}{\vdash C, C^\perp}\,(ax)}{\vdash A \otimes B, A^\perp, B^\perp, C, C^\perp}\,(mix_2)$$

$$\dfrac{\dfrac{\vdash A \otimes B, A^\perp, B^\perp, C, C^\perp}{\vdash A \otimes B, A^\perp \parr B^\perp, C, C^\perp}\,(\parr)}{\vdash A \otimes B, (A^\perp \parr B^\perp) \parr C, C^\perp}\,(\parr)$$



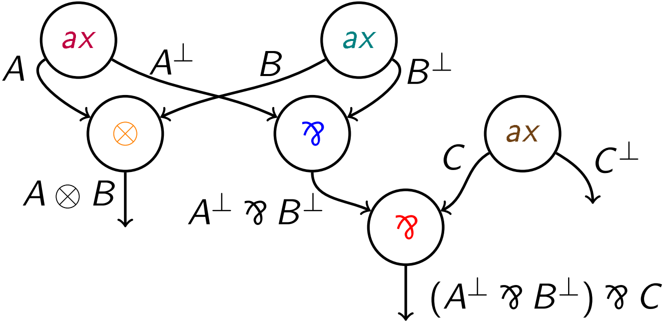
Example of proof structure by desequentialization

$$\dfrac{\dfrac{\dfrac{\dfrac{\overline{\vdash A^{\perp}, A}\ (ax) \qquad \overline{\vdash B, B^{\perp}}\ (ax)}{\vdash A \otimes B, A^{\perp}, B^{\perp}}\ (\otimes) \qquad \overline{\vdash C, C^{\perp}}\ (ax)}{\vdash A \otimes B, A^{\perp}, B^{\perp}, C, C^{\perp}}\ (mix_2)}{\vdash A \otimes B, A^{\perp} \parr B^{\perp}, C, C^{\perp}}\ (\parr)}{\vdash A \otimes B, (A^{\perp} \parr B^{\perp}) \parr C, C^{\perp}}\ (\parr)$$



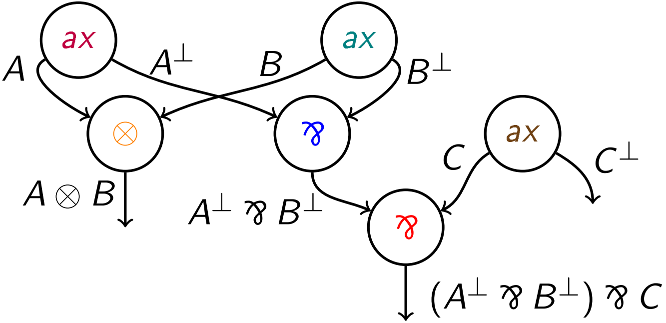
Example of proof structure by desequentialization

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \qquad \dfrac{}{\vdash B, B^\perp}\,(ax)}{\vdash A \otimes B, A^\perp, B^\perp}\,(\otimes) \qquad \dfrac{}{\vdash C, C^\perp}\,(ax)}{\vdash A \otimes B, A^\perp, B^\perp, C, C^\perp}\,(mix_2)}{\vdash A \otimes B, A^\perp \parr B^\perp, C, C^\perp}\,(\parr)}{\vdash A \otimes B, (A^\perp \parr B^\perp) \parr C, C^\perp}\,(\parr)$$



Example of proof structure by desequentialization

$$
\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \qquad \dfrac{\dfrac{}{\vdash B, B^\perp}\,(ax) \qquad \dfrac{}{\vdash C, C^\perp}\,(ax)}{\vdash B, B^\perp, C, C^\perp}\,(mix_2)}{\vdash A \otimes B, A^\perp, B^\perp, C, C^\perp}\,(\otimes)}{\vdash A \otimes B, A^\perp \parr B^\perp, C, C^\perp}\,(\parr)}{\vdash A \otimes B, (A^\perp \parr B^\perp) \parr C, C^\perp}\,(\parr)
$$



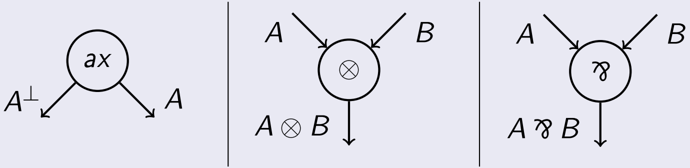
Proof structure

Definition

Partial labeled directed multigraph

vertices $\rightarrow$ ax / $\otimes$ / $⅋$

edges $\rightarrow$ formula



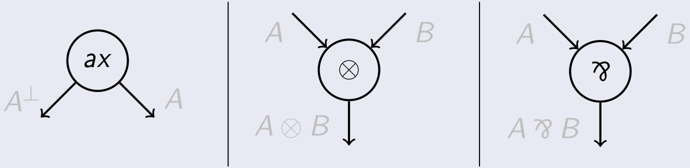
Proof structure

Definition

Partial labeled directed multigraph

vertices $\rightarrow$ ax / $\otimes$ / $⅋$

edges $\rightarrow$ formula

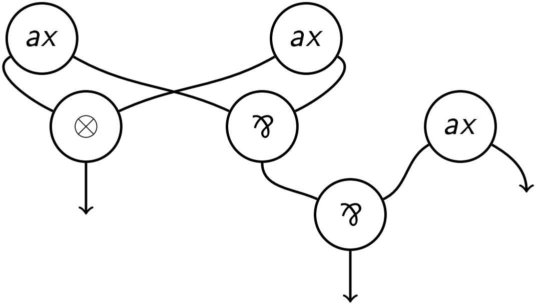


We will not care about edge labels.

Vocabulary on paths

We consider *non-oriented simple* paths.

Definition

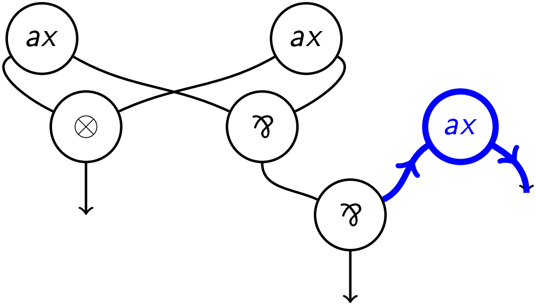


Vocabulary on paths

We consider *non-oriented simple* paths.

Definition

- *Switching* path: does not contain the two premises of any ⅋

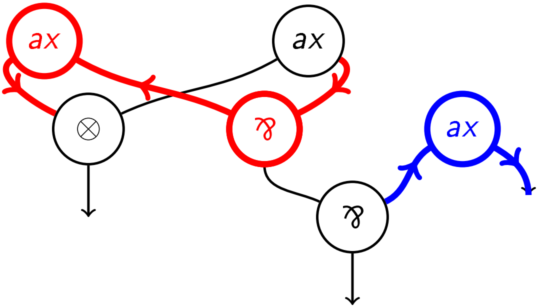


Vocabulary on paths

We consider *non-oriented simple* paths.

Definition

- *Switching* path: does not contain the two premises of any ⅋
- *Strong* path: does not start from a ⅋ by one of its premises

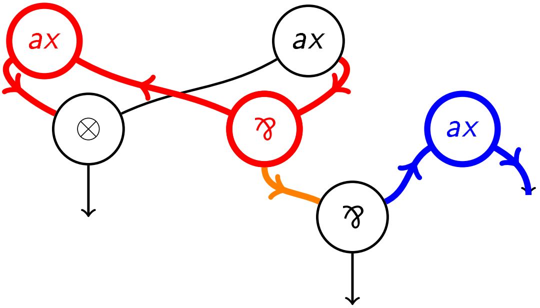


Vocabulary on paths

We consider *non-oriented simple* paths.

Definition

- *Switching* path: does not contain the two premises of any ⅋
- *Strong* path: does not start from a ⅋ by one of its premises
- *Strong-weak* path: strong and ends on a ⅋ with one of its premises

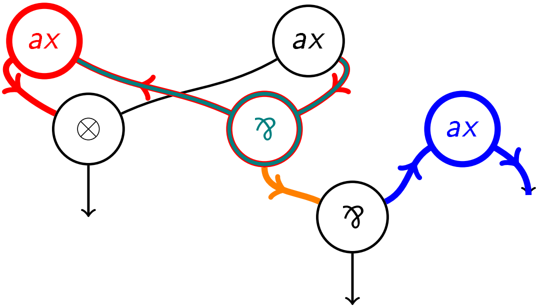


Vocabulary on paths

We consider *non-oriented simple* paths.

Definition

- *Switching* path: does not contain the two premises of any ⅋
- *Strong* path: does not start from a ⅋ by one of its premises
- *Strong-weak* path: strong and ends on a ⅋ with one of its premises
- *Bridge*: pair of consecutive premises of a ⅋ w; w is the bridge *pier*

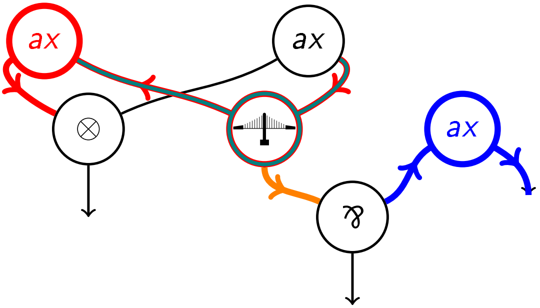


Vocabulary on paths

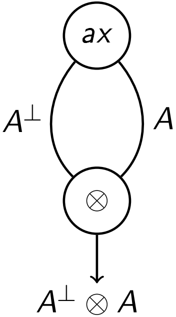
We consider *non-oriented simple* paths.

Definition

- *Switching* path: does not contain the two premises of any ⅋
- *Strong* path: does not start from a ⅋ by one of its premises
- *Strong-weak* path: strong and ends on a ⅋ with one of its premises
- *Bridge*: pair of consecutive premises of a ⅋ w; w is the bridge *pier*



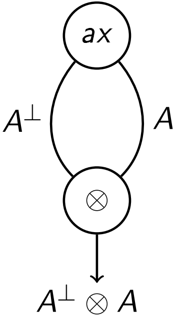
Correctness



Correctness

Danos-Regnier Correctness Criterion

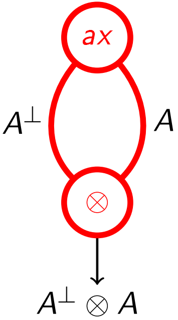
A proof structure is *correct* if it does not contain any switching cycle.



Correctness

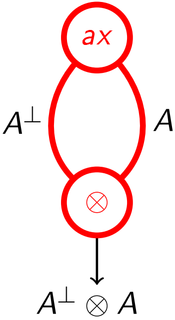
Danos-Regnier Correctness Criterion

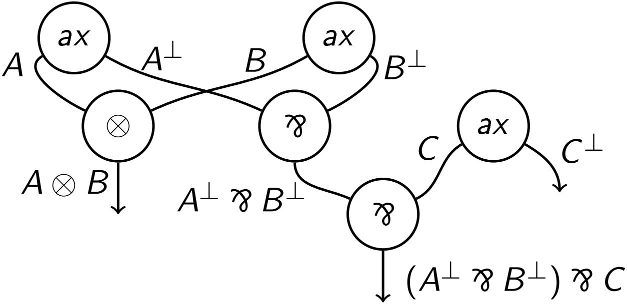
A proof structure is *correct* if it does not contain any switching cycle.



Danos-Regnier Correctness Criterion

A proof structure is *correct* if it does not contain any switching cycle.

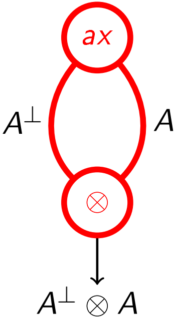


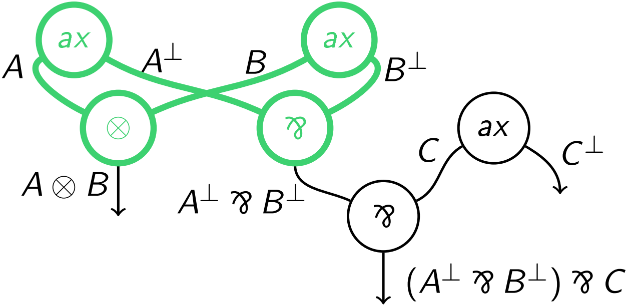


Correctness

Danos-Regnier Correctness Criterion

A proof structure is *correct* if it does not contain any switching cycle.

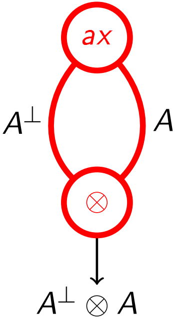


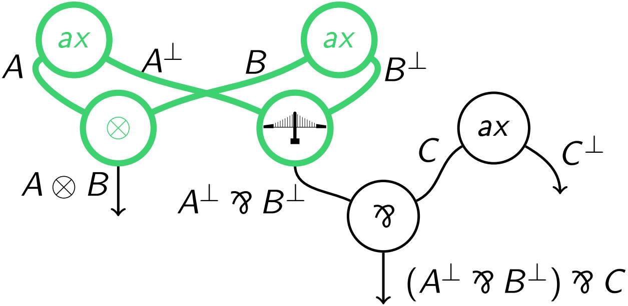


Correctness

Danos-Regnier Correctness Criterion

A proof structure is *correct* if it does not contain any switching cycle.





Outline

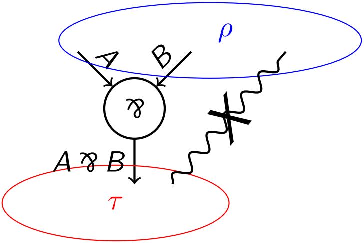
Destination Sequentialization

Sequentialization

Given a correct proof structure, there is a proof desequentializing to it.

Splitting ⅋-node

A ⅋-node is *splitting* if there is no cycle containing its conclusion.



Order $\prec$ on $⅋$-nodes

Definition

$v \prec u$ means v and u are distinct $⅋$-nodes and there is a path p such that:

(1) p is a strong-weak bridge-free path from v to u

(2) there is no strong bridge-free path starting from u and going back on p

Order $\prec$ on $\parr$-nodes

Definition

$v \prec u$ means v and u are distinct $\parr$-nodes and there is a path p such that:

(1) p is a strong-weak bridge-free path from v to u

(2) there is no strong bridge-free path starting from u and going back on p

Lemma

$\prec$ is a strict partial order relation.

Proof.

Irreflexivity: by definition. Transitivity: assume $v \prec u \prec w$.

(1) ?

(2) ?

□

Order $\prec$ on ⅋-nodes

Definition

$v \prec u$ means v and u are distinct ⅋-nodes and there is a path p such that:

(1) p is a strong-weak bridge-free path from v to u

(2) there is no strong bridge-free path starting from u and going back on p

Lemma

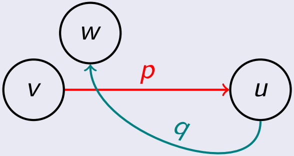
$\prec$ is a strict partial order relation.

Proof.

Irreflexivity: by definition. Transitivity: assume $v \prec u \prec w$.

(1) ?

(2) ?



□

Order $\prec$ on ⅋-nodes

Definition

$v \prec u$ means v and u are distinct ⅋-nodes and there is a path p such that:

(1) p is a strong-weak bridge-free path from v to u

(2) there is no strong bridge-free path starting from u and going back on p

Lemma

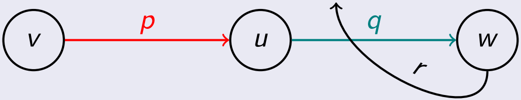
$\prec$ is a strict partial order relation.

Proof.

Irreflexivity: by definition. Transitivity: assume $v \prec u \prec w$.

(1) $\surd$

(2) ?



□

Order $\prec$ on ⅋-nodes

Definition

$v \prec u$ means v and u are distinct ⅋-nodes and there is a path p such that:

(1) p is a strong-weak bridge-free path from v to u

(2) there is no strong bridge-free path starting from u and going back on p

Lemma

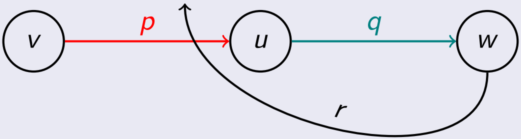
$\prec$ is a strict partial order relation.

Proof.

Irreflexivity: by definition. Transitivity: assume $v \prec u \prec w$.

(1) $\surd$

(2) $\surd$



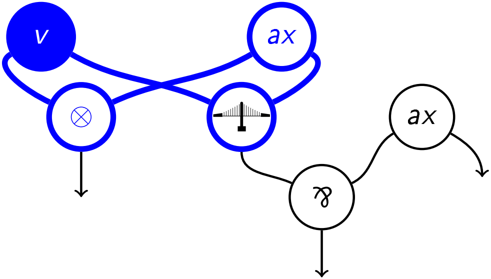
□

Bungee Jumping

Definition

For a vertex v, $\mathcal{M}_v$ is the set of cycles with source v, containing a conclusion of v, and with $\underbrace{\text{a minimal number of bridges}}_{\geq 1 \text{ by correctness}}$ among such cycles.

For v a $\parr$-node, $\mathcal{M}_v = \emptyset$ if and only if v is splitting.

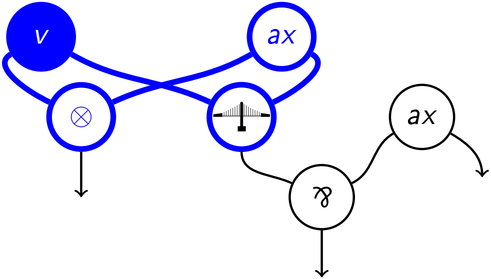


Bungee Jumping

Definition

For a vertex v, $\mathcal{M}_v$ is the set of cycles with source v, containing a conclusion of v, and with $\underbrace{\text{a minimal number of bridges}}_{\geq 1 \text{ by correctness}}$ among such cycles.

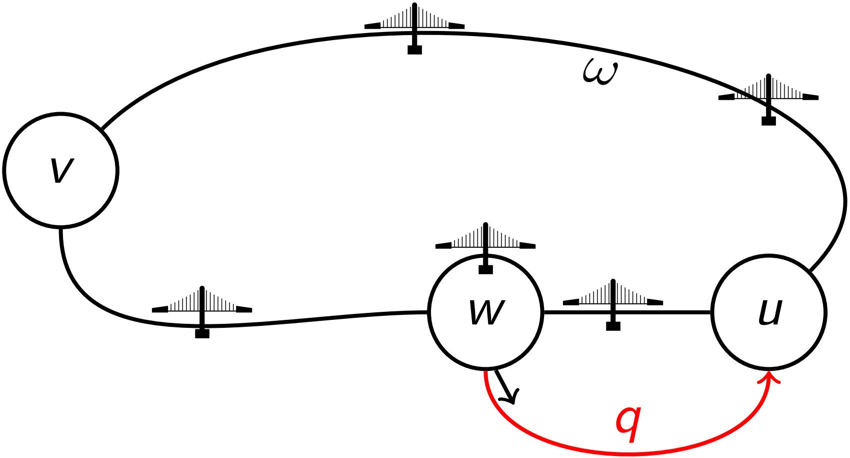
For v a ⅋-node, $\mathcal{M}_v = \emptyset$ if and only if v is splitting.



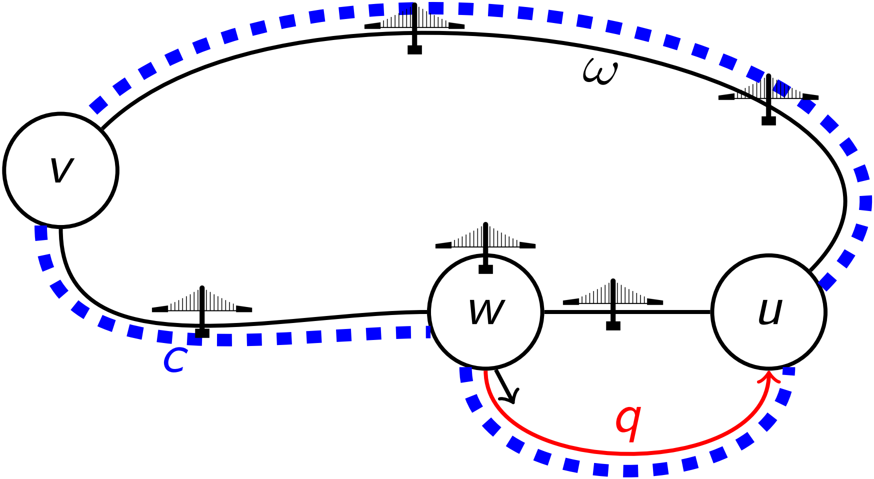
Bungee Jumping

Let ω be a cycle in $\mathcal{M}_v$. There is no strong bridge-free path q with source w the pier of a bridge of ω and going back on ω.

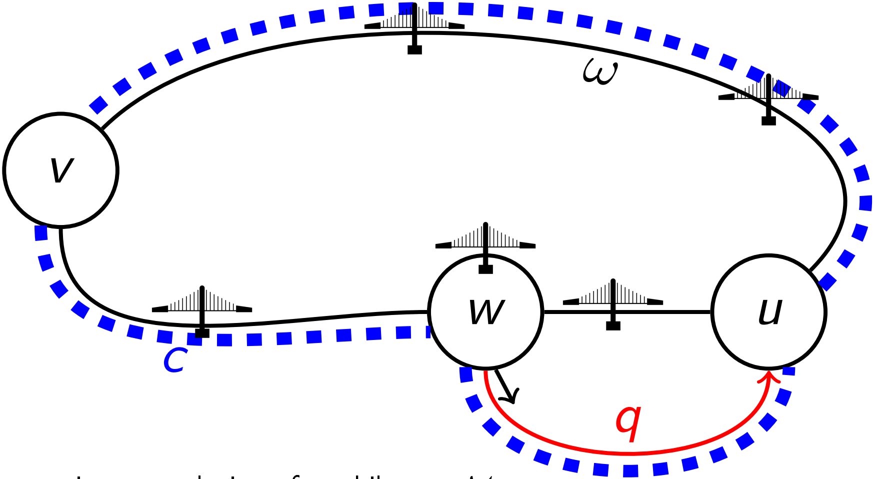
Proof of Bungee Jumping



Proof of Bungee Jumping



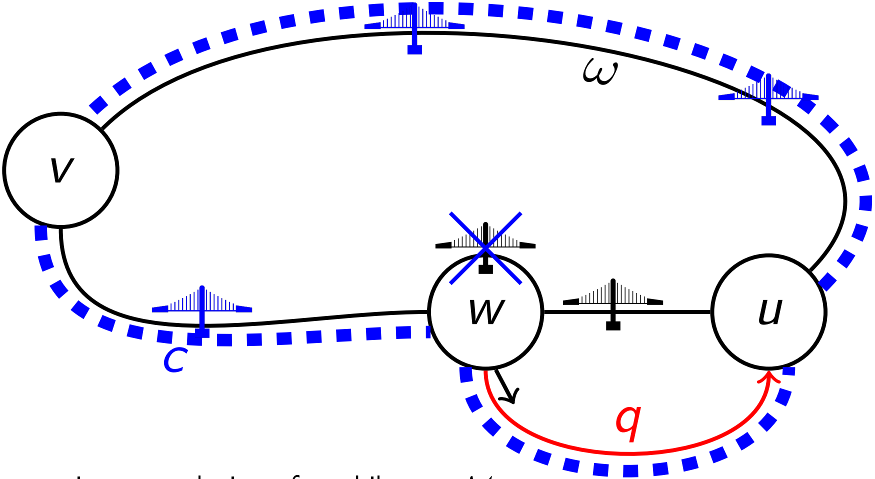
Proof of Bungee Jumping



c contains a conclusion of v while $\omega \in \mathcal{M}_v$

$\implies$ c has at least as many bridges as ω

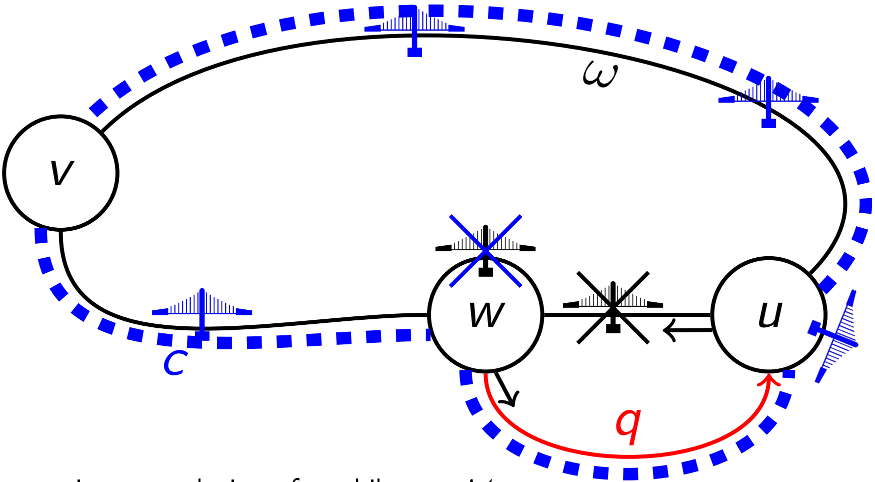
Proof of Bungee Jumping



c contains a conclusion of v while $\omega \in \mathcal{M}_v$

$\Longrightarrow$ c has at least as many bridges as ω

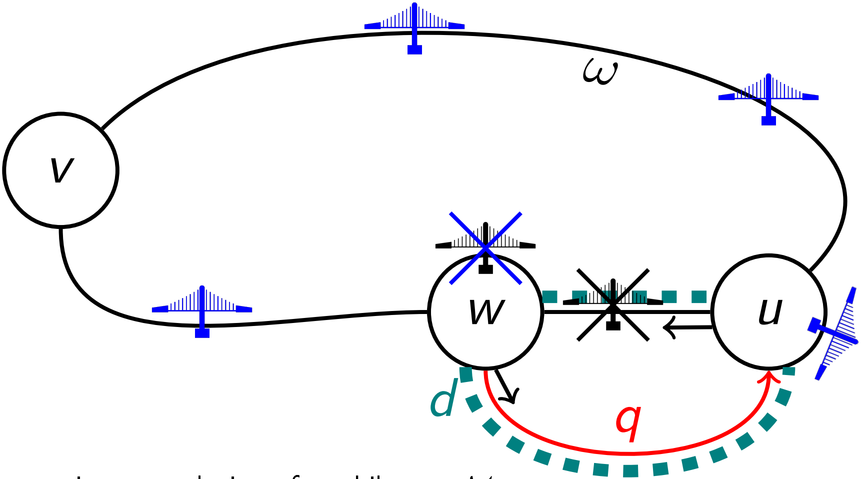
Proof of Bungee Jumping



c contains a conclusion of v while $\omega \in \mathcal{M}_v$

$\implies$ c has at least as many bridges as ω

Proof of Bungee Jumping



c contains a conclusion of v while $\omega \in \mathcal{M}_v$

$\implies$ c has at least as many bridges as ω

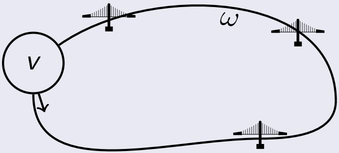
Splitting ⅋

Lemma

Let v be a non-splitting ⅋-node. There exists w such that $v \prec w$.

Proof.

Take $\omega \in \mathcal{M}_v \neq \emptyset$.



□

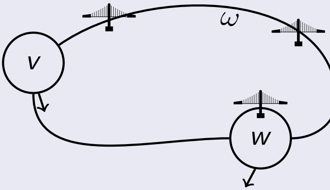
Splitting ⅋

Lemma

Let v be a non-splitting ⅋-node. There exists w such that $v \prec w$.

Proof.

Take $\omega \in \mathcal{M}_v \neq \emptyset$.
It contains some pier: set w the **first** one going from the conclusion of v.



□

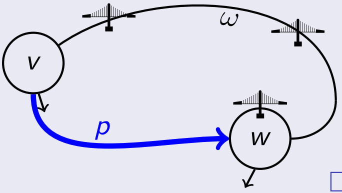
Splitting ⅋

Lemma

Let v be a non-splitting ⅋-node. There exists w such that $v \prec w$.

Proof.

Take $\omega \in \mathcal{M}_v \neq \emptyset$.
It contains some pier: set w the **first** one going from the conclusion of v.



□

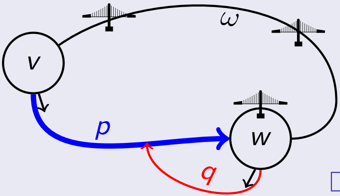
Splitting ⅋

Lemma

Let v be a non-splitting ⅋-node. There exists w such that $v \prec w$.

Proof.

Take $\omega \in \mathcal{M}_v \neq \emptyset$.
It contains some pier: set w the **first** one going from the conclusion of v.
By Bungee Jumping, $v \prec w$.



□

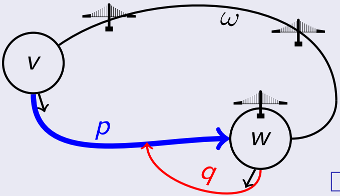
Splitting ⅋

Lemma

Let v be a non-splitting ⅋-node. There exists w such that $v \prec w$.

Proof.

Take $\omega \in \mathcal{M}_v \neq \emptyset$.
It contains some pier: set w the **first** one going from the conclusion of v.
By Bungee Jumping, $v \prec w$.



□

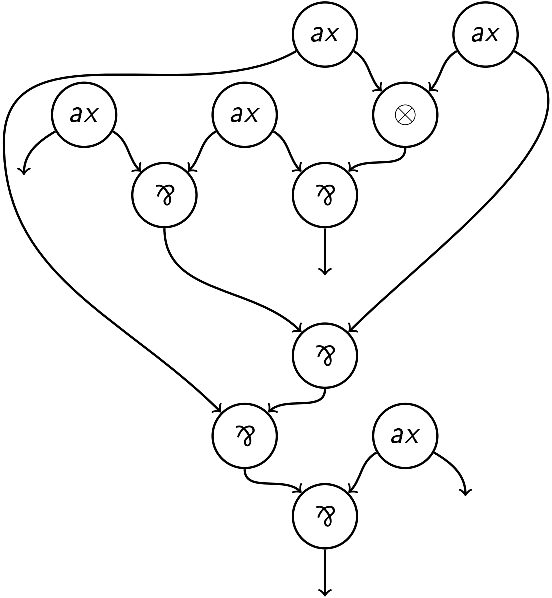
Splitting ⅋

A correct proof structure is ⅋-free or contains a splitting ⅋-node.

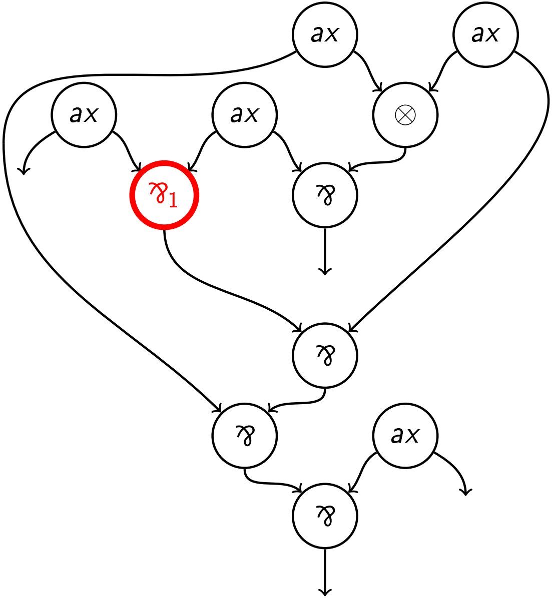
Proof.

A maximal ⅋ for the strict partial order $\prec$ is splitting. □

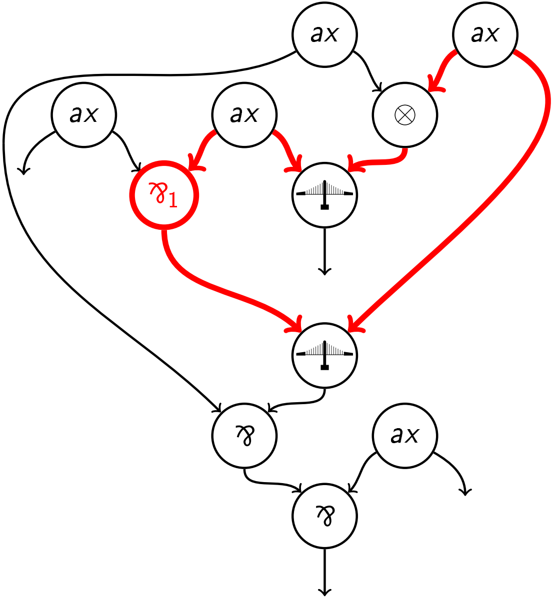
Finding a splitting ⅋ on an example



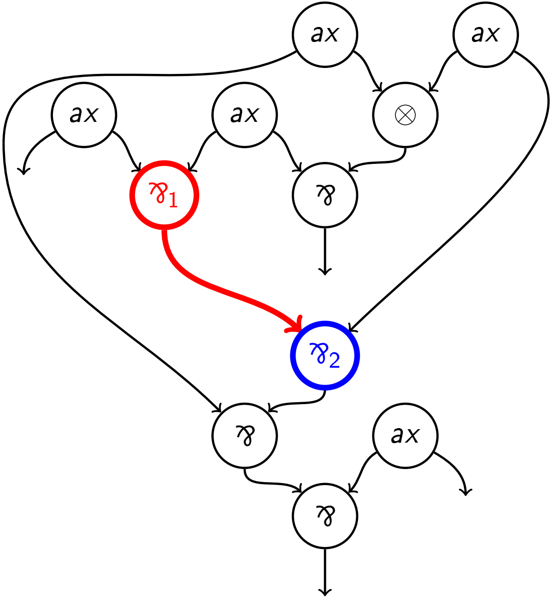
Finding a splitting ⅋ on an example



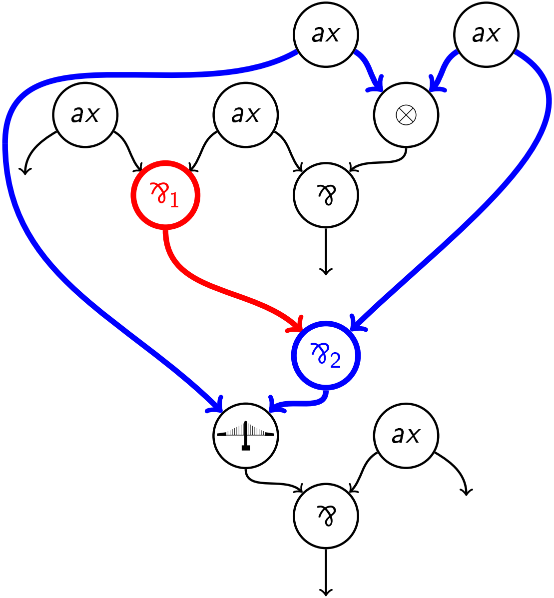
Finding a splitting ⅋ on an example



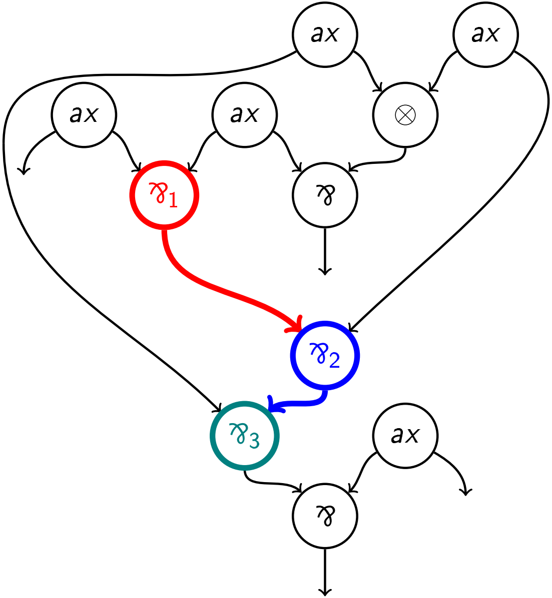
Finding a splitting ⅋ on an example



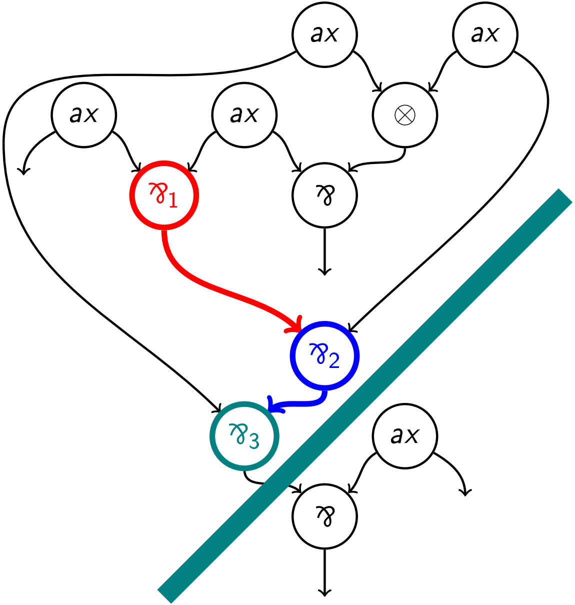
Finding a splitting ⅋ on an example



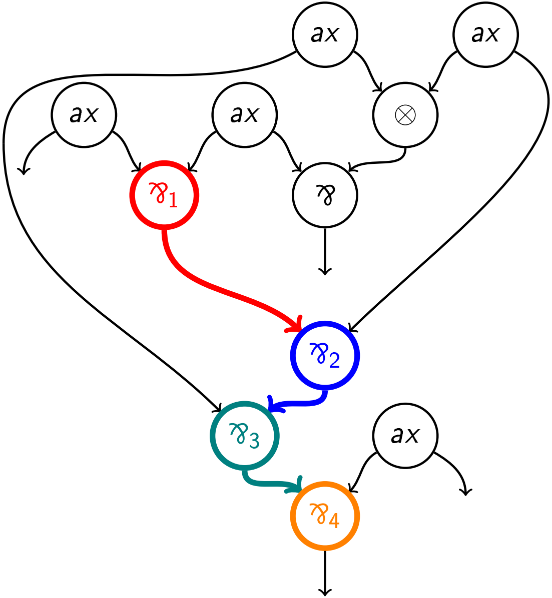
Finding a splitting ⅋ on an example



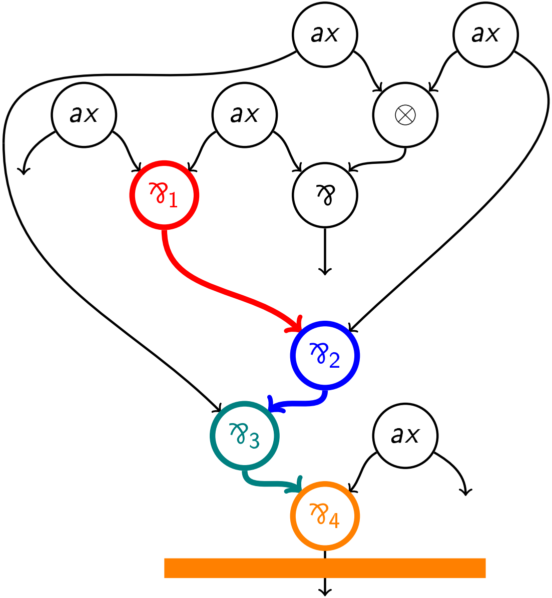
Finding a splitting ⅋ on an example



Finding a splitting ⅋ on an example



Finding a splitting ⅋ on an example



Conclusion

Sequentialization

Given a correct proof structure, there is a proof desequentializing to it.

Conclusion

Sequentialization

Given a correct proof structure, there is a proof desequentializing to it.

- Can deduct the result without the *mix*-rules from this one.

$$\#cc = 1 + \#mix_2 - \#mix_0$$

Conclusion

Sequentialization

Given a correct proof structure, there is a proof desequentializing to it.

- Can deduct the result without the *mix*-rules from this one.

$$\#cc = 1 + \#mix_2 - \#mix_0$$

- Splitting ⅋ formalized in Coq.

Conclusion

Sequentialization

Given a correct proof structure, there is a proof desequentializing to it.

- Can deduct the result without the *mix*-rules from this one.

$$\#cc = 1 + \#mix_2 - \#mix_0$$

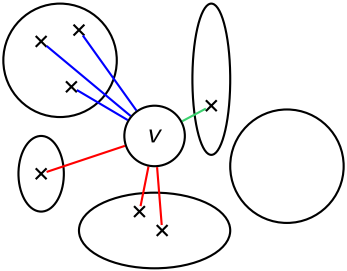
- Splitting $\parr$ formalized in Coq.
- Can be extended to proof nets with additives (from Hugues & Van Glabbeek [HvG05]).

Equivalent in graph theory [Ngu20]

This proof can be generalized to colored graphs.
Alternating cycle: with consecutive edges of different colors

Yeo's Theorem [Yeo97]

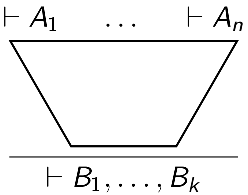
Let G be a (non-empty) edge-colored unoriented graph ***with no alternating cycle****. There exists a vertex v of G such that no component of $G - v$ is joint to v with edges of more than one color.*



Thank you!

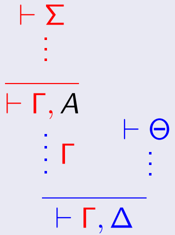
Open proofs

$$\frac{}{\vdash A}\ (hyp)$$

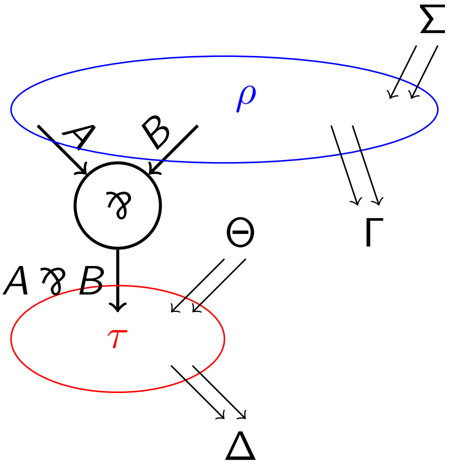


Substitution

$\pi_2 [\pi_1 / A] =$



Sequentialization with hypotheses



By induction:

- $\rho \to \pi$
- $\tau \to \sigma$

Proof of the whole structure:

$$\sigma\left[\frac{\pi}{\vdash A \parr B, \Gamma}\ (\parr) \Big/ A \parr B\right]$$

References

Dominic Hughes and Rob van Glabbeek.
Proof nets for unit-free multiplicative-additive linear logic.
ACM Transactions on Computational Logic, 6(4):784–842, 2005.

Lê Thành Dũng Nguyễn.
Unique perfect matchings, forbidden transitions and proof nets for linear logic with mix.
Logical Methods in Computer Science, 16(1), February 2020.

Anders Yeo.
A note on alternating cycles in edge-coloured graphs.
Journal of Combinatorial Theory, Series B, 69(2):222–225, 1997.