Sublogarithmic uniform Boolean Proof Nets

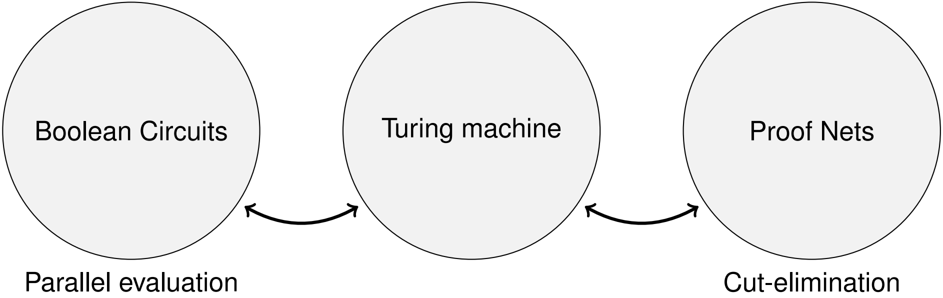
DICE 2011 Second Workshop on Developments in Implicit Computational Complexity

Clément Aubert
aubert@lipn.univ-paris13.fr

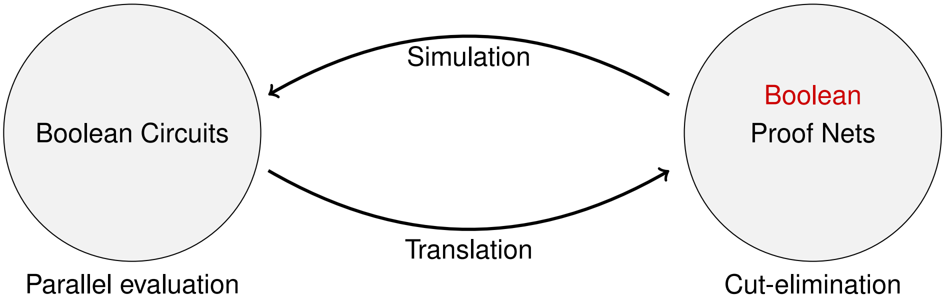
Institut Galilée - Université Paris-Nord
99, avenue Jean-Baptiste Clément
93430 Villetaneuse

2nd-3rd April 2011, Saarbrücken

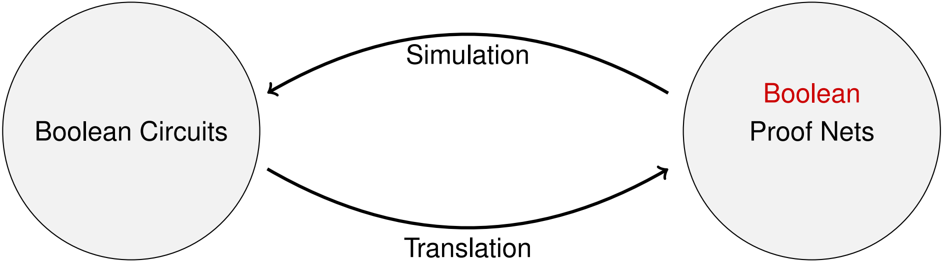
Two models of parallel computation



Two models of parallel computation



Two models of parallel computation



Parallel evaluation

Cut-elimination

Implicit complexity framework:

- Size and depth
- Resources needed by the reductions
- Uniformity

Previous works and new results

Previous works

- [Terui, 2004] did not took into account uniformity.
- [Mogbil and Rahli, 2007] did, but translation was in L.

Previous works and new results

Previous works

- [Terui, 2004] did not took into account uniformity.
- [Mogbil and Rahli, 2007] did, but translation was in L.

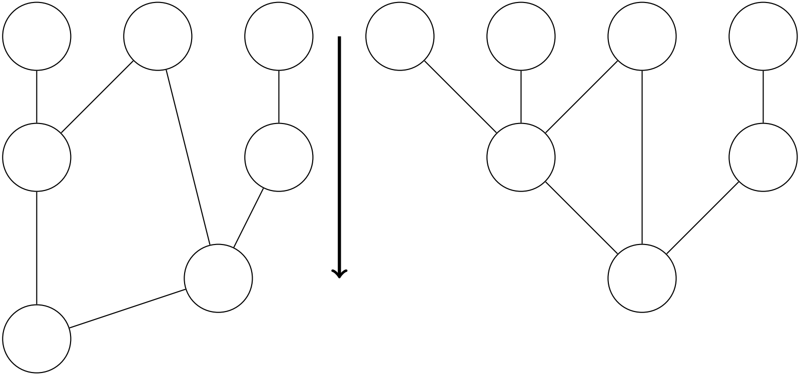
Results

- *Proof Circuits lighten the size and translate all kind of Boolean Circuits.*
- *Translation in AC^0 concerning uniform families.*
- *Proof Nets compute functions under L.*

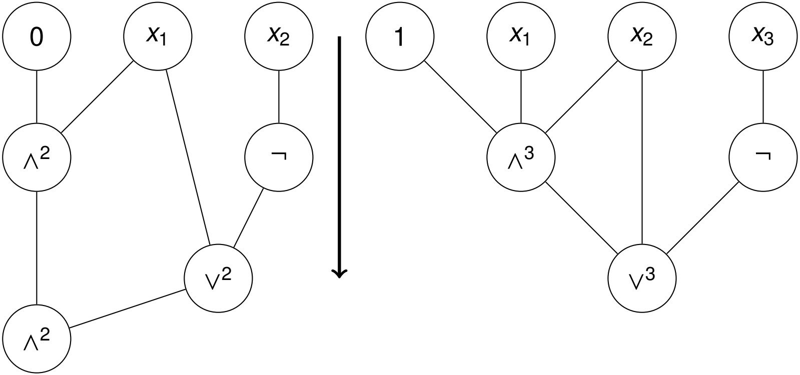
Map

- First definitions: Boolean Circuits and Boolean Proof Nets
- Simulation of Boolean Proof Nets by Boolean Circuits
- Examples of translation and composition
- Proof Circuits and translation

Examples (Two Boolean Circuits)



Examples (Two Boolean Circuits)



Definition (Basis)

$\mathfrak{B}_0 = \{\neg, \wedge^2, \vee^2\}$
$\mathfrak{B}_1 = \{\neg, (\wedge^n)_{n \geqslant 2}, (\vee^n)_{n \geqslant 2}\}.$

Definition (NC^i resp. AC^i)

For $i \in \mathbb{N}$, set of boolean functions computable by a uniform family of Boolean Circuits $C = (C_n)$ where for all C_n

- *its depth is $O(\log^i n)$,*
- *its size is polynomial in n,*
- *its gate are labeled with functions of $\mathfrak{B}_0$ resp. $\mathfrak{B}_1$.*

$$NC = \bigcup_{i \in \mathbb{N}} NC^i = \bigcup_{i \in \mathbb{N}} AC^i = AC$$

Complexity Classes: *NC* and *AC*

Definition (NC^i resp. AC^i)

For $i \in \mathbb{N}$, set of boolean functions computable by a uniform family of Boolean Circuits $C = (C_n)$ where for all C_n

- *its depth is $O(\log^i n)$,*
- *its size is polynomial in n,*
- *its gate are labeled with functions of $\mathfrak{B}_0$ resp. $\mathfrak{B}_1$.*

$$NC = \bigcup_{i \in \mathbb{N}} NC^i = \bigcup_{i \in \mathbb{N}} AC^i = AC$$

Theorems

$$AC^0 \subsetneq NC^1 \subseteq L \subseteq NL \subseteq AC^1 \subseteq NC^1 \subseteq AC^2 \subseteq \ldots$$

$$AC^0(UstCONN_2) \subseteq AC^1 \supseteq L$$

Definition (Rules of **MLLu**)

$$\frac{}{\vdash A, A^\perp}\, ax. \qquad \frac{\vdash \Gamma_1, A_1 \quad \dots \quad \vdash \Gamma_n, A_n}{\vdash \Gamma_1, \dots, \Gamma_n, \otimes^n(A_1, \dots, A_n)}\, \otimes^n$$

$$\frac{\vdash \Gamma, A \qquad \vdash \Delta, A^\perp}{\vdash \Gamma, \Delta}\, cut \qquad \frac{\vdash \Gamma, A_n, \dots, A_1}{\vdash \Gamma, \invamp^n(A_n, \dots, A_1)}\, \invamp^n$$

Definition (Boolean Type, [Terui, 2004])

$$\dfrac{\dfrac{\dfrac{}{\vdash \alpha^\perp, \alpha}\, ax. \qquad \dfrac{}{\vdash \alpha^\perp, \alpha}\, ax.}{\vdash \alpha^\perp, \alpha^\perp, \alpha \otimes \alpha}\, \otimes^2}{\vdash \invamp^3(\alpha^\perp, \alpha^\perp, \alpha \otimes \alpha)}\, \invamp^3$$

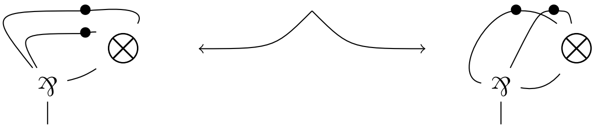
Definition (Rules of **MLLu**)

$$\frac{}{\vdash A, A^{\perp}}\ ax. \qquad \frac{\vdash \Gamma_1, A_1 \quad \ldots \quad \vdash \Gamma_n, A_n}{\vdash \Gamma_1, \ldots, \Gamma_n, \otimes^n(A_1, \ldots, A_n)}\ \otimes^n$$

$$\frac{\vdash \Gamma, A \quad \vdash \Delta, A^{\perp}}{\vdash \Gamma, \Delta}\ cut \qquad \frac{\vdash \Gamma, A_n, \ldots, A_1}{\vdash \Gamma, \parr^n(A_n, \ldots, A_1)}\ \parr^n$$

Definition (Boolean Type, [Terui, 2004])

$$\cfrac{\cfrac{\cfrac{}{\vdash \alpha^{\perp}, \alpha}\ ax. \qquad \cfrac{}{\vdash \alpha^{\perp}, \alpha}\ ax.}{\vdash \alpha^{\perp}, \alpha^{\perp}, \alpha \otimes \alpha}\ \otimes^2}{\vdash \parr^3(\alpha^{\perp}, \alpha^{\perp}, \alpha \otimes \alpha)}\ \parr^3$$



MLLu

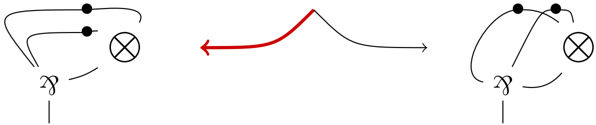
Definition (Rules of **MLLu**)

$$\frac{}{\vdash A, A^{\perp}}\ ax. \qquad \frac{\vdash \Gamma_1, A_1 \quad \ldots \quad \vdash \Gamma_n, A_n}{\vdash \Gamma_1, \ldots, \Gamma_n, \otimes^n(A_1, \ldots, A_n)}\ \otimes^n$$

$$\frac{\vdash \Gamma, A \qquad \vdash \Delta, A^{\perp}}{\vdash \Gamma, \Delta}\ cut \qquad \frac{\vdash \Gamma, A_n, \ldots, A_1}{\vdash \Gamma, \parr^n(A_n, \ldots, A_1)}\ \parr^n$$

Definition (Boolean Type, [Terui, 2004])

$$\dfrac{\dfrac{\dfrac{}{\vdash p : \alpha^{\perp}, p : \alpha \rhd ax_p}\ ax. \qquad \dfrac{}{\vdash q : \alpha^{\perp}, q : \alpha \rhd ax_q}\ ax.}{\vdash p : \alpha^{\perp}, q : \alpha^{\perp}, r : \alpha \otimes \alpha \rhd tensor_r^{p,q}(ax_p, ax_q)}\ \otimes^2}{\vdash s : \parr^3(\alpha^{\perp}, \alpha^{\perp}, \alpha \otimes \alpha) \rhd par_s^{q,p,r}(tensor_r^{p,q}(ax_p, ax_q))}\ \parr^3$$

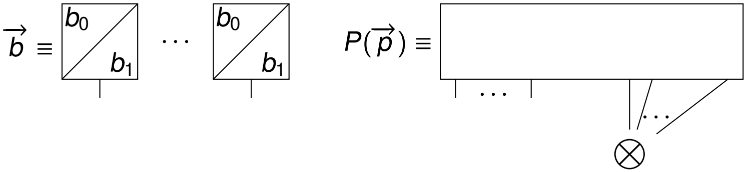


Definition (Boolean Proof Nets, [Terui, 2004])

A Boolean Proof Net $P(\overrightarrow{p})$ with n inputs is a Proof Net of type

$$\vdash p_1 : \boldsymbol{B}^{\perp}[A_1], \ldots, p_n : \boldsymbol{B}^{\perp}[A_n], s : \otimes^{1+m}(\boldsymbol{B}[A], C_1, \ldots, C_m)$$

Definition ($P(\overrightarrow{b})$)

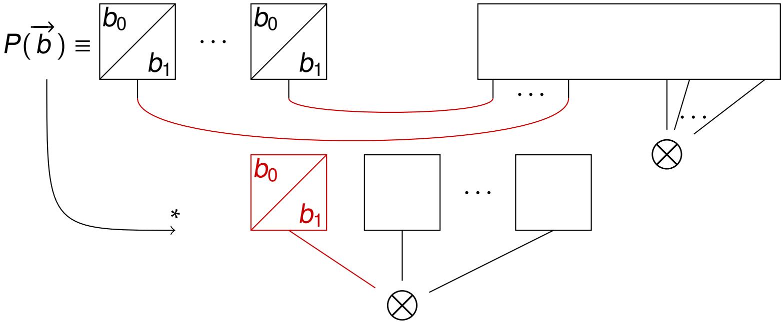


Definition (Boolean Proof Nets, [Terui, 2004])

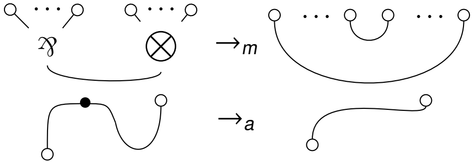
A Boolean Proof Net $P(\overrightarrow{p})$ with n inputs is a Proof Net of type

$$\vdash p_1 : \boldsymbol{B}^{\perp}[A_1], \ldots, p_n : \boldsymbol{B}^{\perp}[A_n], s : \otimes^{1+m}(\boldsymbol{B}[A], C_1, \ldots, C_m)$$

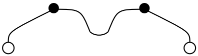
Definition ($P(\overrightarrow{b})$)



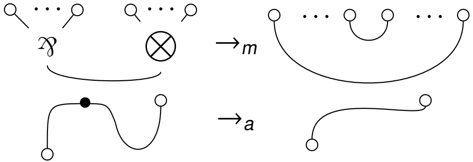
Definition ($\rightarrow_m$ and $\rightarrow_a$)



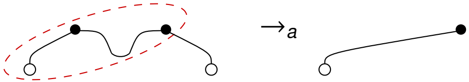
Remark (Critical pair)



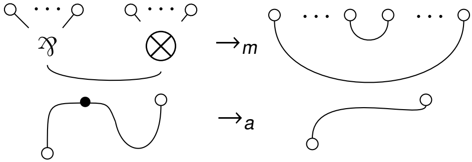
Definition ($\to_m$ and $\to_a$)



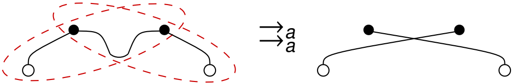
Remark (Critical pair)



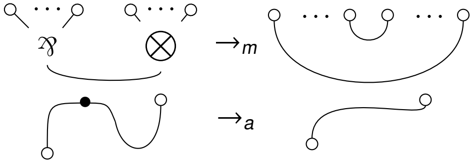
Definition ($\to_m$ and $\to_a$)



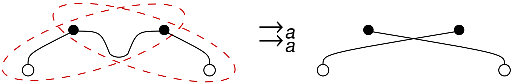
Remark (Critical pair)



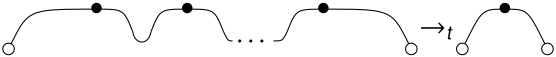
Definition ($\to_m$ and $\to_a$)



Remark (Critical pair)



Definition (t-reduction)



Map

- First definitions: Boolean Circuits and Boolean Proof Nets
- Simulation of Boolean Proof Nets by Boolean Circuits
- Examples of translation and composition
- Proof Circuits and translation

Simulation of $\rightrightarrows$ by Boolean Circuits

Theorem ([Terui, 2004])

Every Proof-Net P normalizes in at most $3 \times d(P)$ steps of parallel reduction ($\rightrightarrows_t$, $\rightrightarrows_a$, $\rightrightarrows_m$).

Simulation of $\rightrightarrows$ by Boolean Circuits

Theorem ([Terui, 2004])

Every Proof-Net P normalizes in at most $3 \times d(P)$ steps of parallel reduction ($\rightrightarrows_t$, $\rightrightarrows_a$, $\rightrightarrows_m$).

Description of a Boolean Proof Net P_n

$\downarrow$

Simulates $\rightrightarrows_t$

Simulates $\rightrightarrows_a$

Simulates $\rightrightarrows_m$

Simulation of $\rightrightarrows$ by Boolean Circuits

Theorem ([Terui, 2004])

Every Proof-Net P normalizes in at most $3 \times d(P)$ steps of parallel reduction ($\rightrightarrows_t$, $\rightrightarrows_a$, $\rightrightarrows_m$).

Description of a Boolean Proof Net P_n

$\downarrow$

$$\left.\begin{array}{c} \boxed{\text{Simulates } \rightrightarrows_t} \\ \boxed{\text{Simulates } \rightrightarrows_a} \\ \boxed{\text{Simulates } \rightrightarrows_m} \end{array}\right\} d(P_n) \text{ times}$$

$\vdots$

Establishes the output of P_n

Simulation of $\rightrightarrows$ by Boolean Circuits

Theorem ([Terui, 2004])

Every Proof-Net P normalizes in at most $3 \times d(P)$ steps of parallel reduction ($\rightrightarrows_t$, $\rightrightarrows_a$, $\rightrightarrows_m$).

Description of a Boolean Proof Net P_n

$\downarrow$

Constant-depth size $O(|P_n|^3)$ $\left\{ \begin{array}{c} \boxed{\text{Simulates } \rightrightarrows_t} \\ \boxed{\text{Simulates } \rightrightarrows_a} \\ \boxed{\text{Simulates } \rightrightarrows_m} \end{array} \right\}$ $d(P_n)$ times

$\vdots$

$\boxed{\text{Establishes the output of } P_n}$

Simulation of $\rightrightarrows$ by Boolean Circuits

Theorem ([Terui, 2004])

Every Proof-Net P normalizes in at most $3 \times d(P)$ steps of parallel reduction ($\rightrightarrows_t$, $\rightrightarrows_a$, $\rightrightarrows_m$).

Description of a Boolean Proof Net P_n

$\downarrow$

Constant-depth size $O(|P_n|^3)$

Simulates $\rightrightarrows_t$

Simulates $\rightrightarrows_a$

Simulates $\rightrightarrows_m$

$d(P_n)$ times

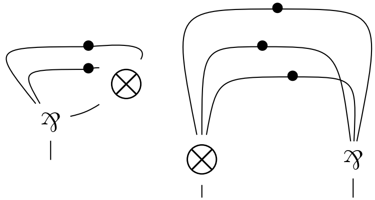
$\vdots$

Establishes the output of P_n

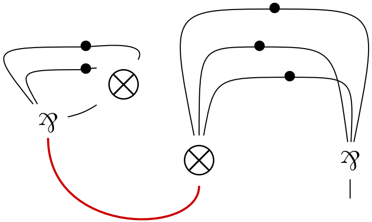
Depth $d(P_n)$ size $O(|P_n|^4)$ over $\mathfrak{B}_1(UstCONN_2)$

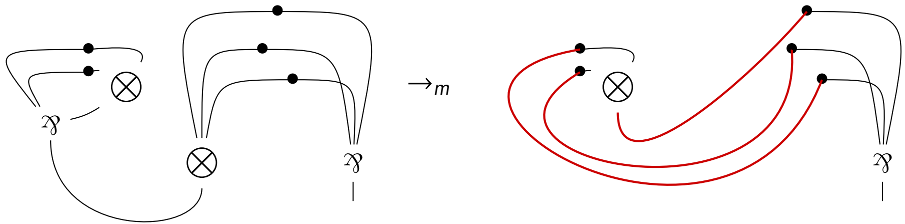
Map

- First definitions: Boolean Circuits and Boolean Proof Nets
- Simulation of Boolean Proof Nets by Boolean Circuits
- Examples of translation and composition
- Proof Circuits and translation

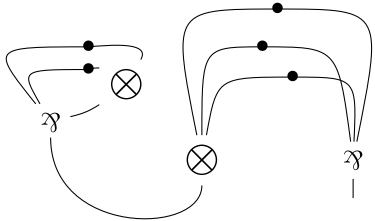


First example: negation

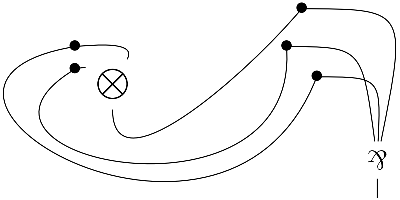



⅋
⅋
$\to_m$
⅋

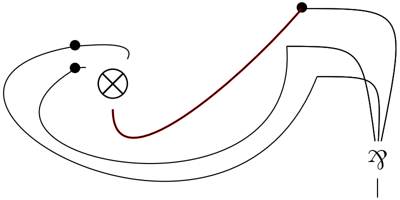
First example: negation



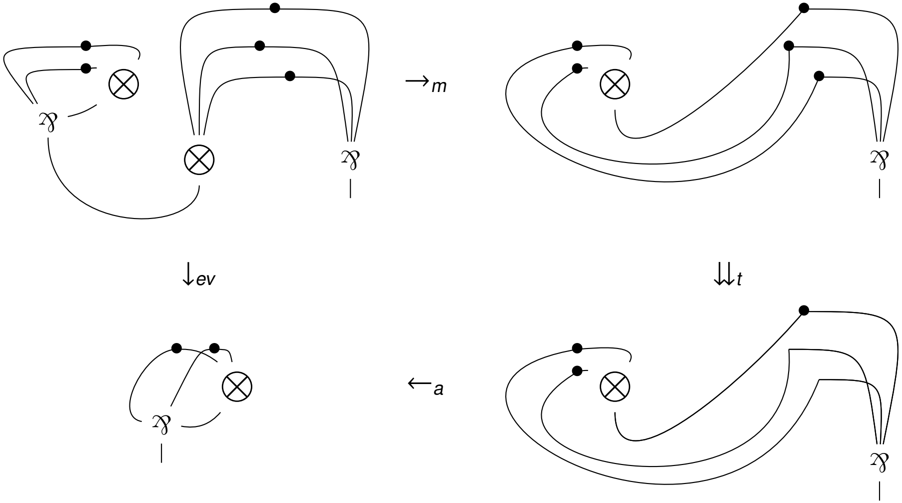
$\rightarrow_m$



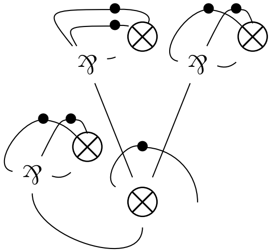
$\downdownarrows_t$



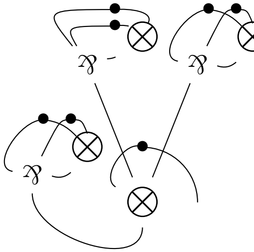
First example: negation



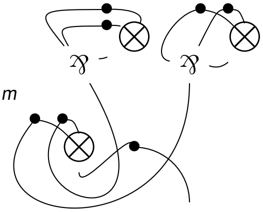
Second example: conditional

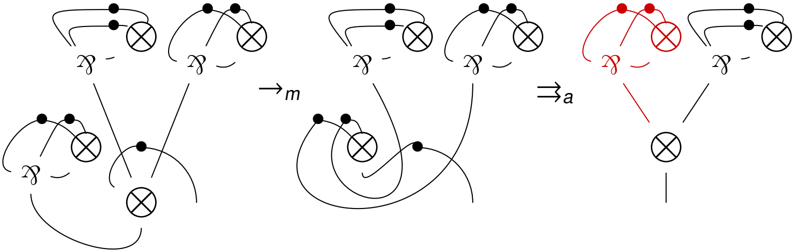


Second example: conditional

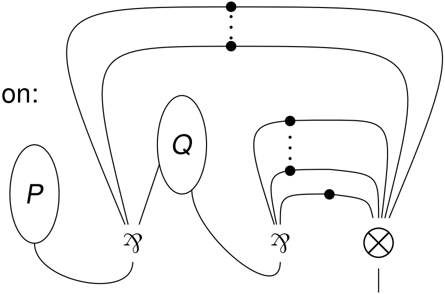


$\rightarrow_m$

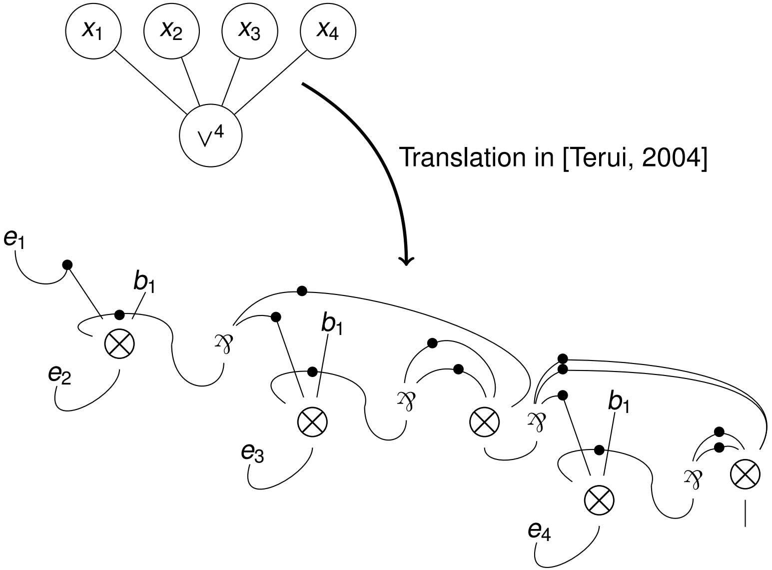




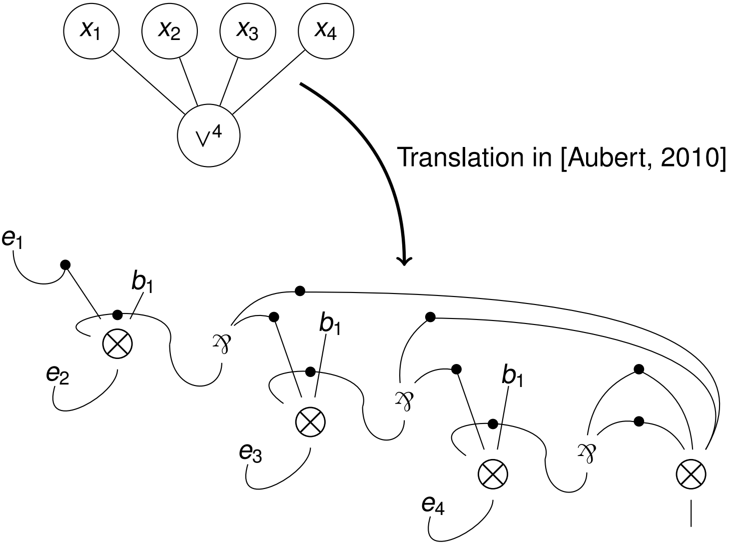
Composition:



$\vee^4$ translated



$\vee^4$ translated

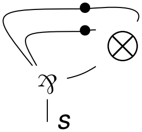


Map

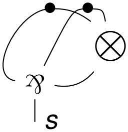
- First definitions: Boolean Circuits and Boolean Proof Nets
- Simulation of Boolean Proof Nets by Boolean Circuits
- Examples of translation and composition
- Proof Circuits and translation

Pieces (1/4)

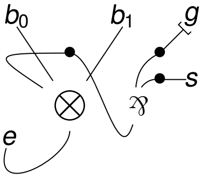
$b_0 \equiv$



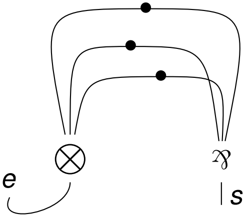
$b_1 \equiv$



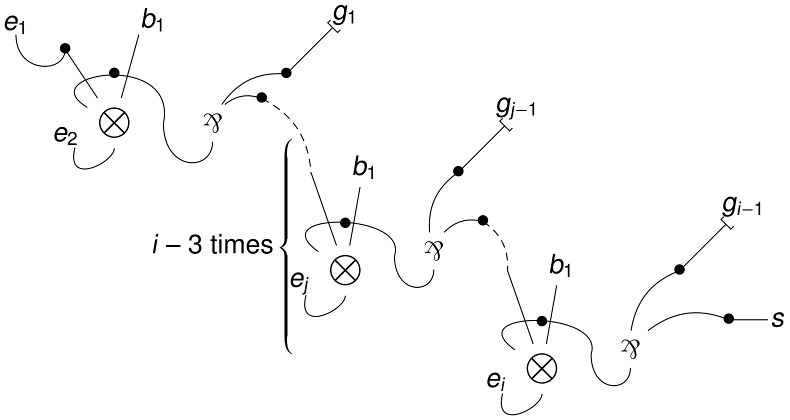
$COND \equiv$



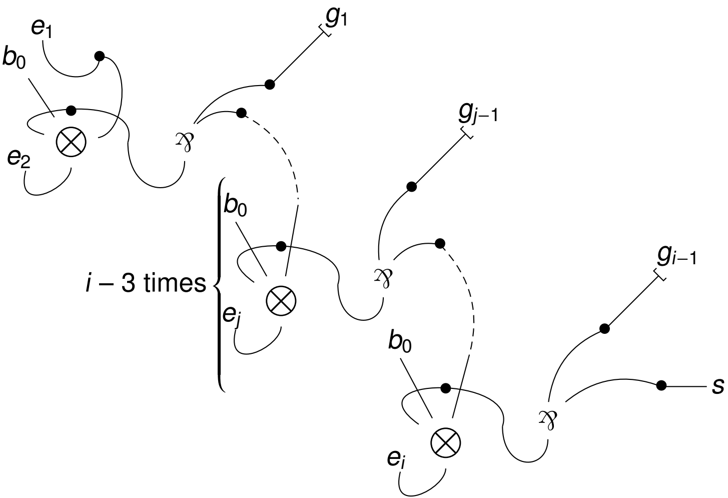
$NEG \equiv$



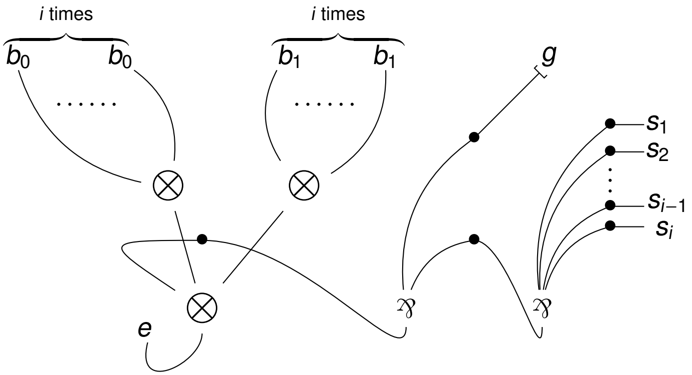
$DISJ^i \equiv$



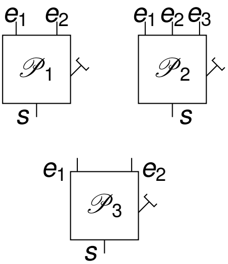
$CONJ^i \equiv$



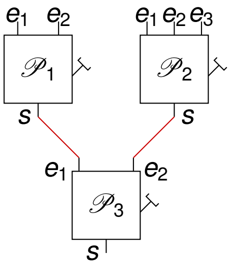
$DUPL^i \equiv$



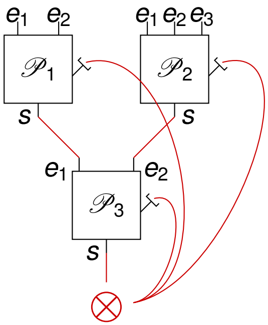
Definition (Proof Circuits)



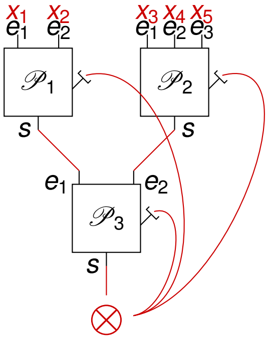
Definition (Proof Circuits)



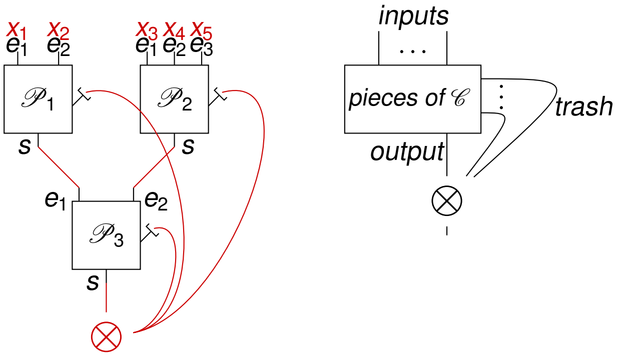
Definition (Proof Circuits)



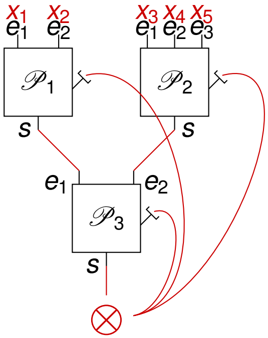
Definition (Proof Circuits)

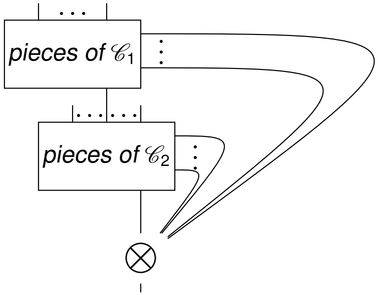


Definition (Proof Circuits)



Definition (Proof Circuits)





Theorem ([Aubert, 2010])

Every Proof Circuits is a Boolean Proof Net.

Complexity Class: *CCP*

Definition (*CCP*^{i} [Aubert, 2010])

For $i \in \mathbb{N}$, set of boolean functions computable by a uniform family of Proof Circuits $P = (P_n)$ where for all P_n

- *its depth is $O(\log^i n)$,*
- *its size is polynomial in n.*

$$CCP = \bigcup_{i \in \mathbb{N}} CCP^i$$

Definition (CCP^i [Aubert, 2010])

For $i \in \mathbb{N}$, set of boolean functions computable by a uniform family of Proof Circuits $P = (P_n)$ where for all P_n

- *its depth is $O(\log^i n)$,*
- *its size is polynomial in n.*

$$CCP = \bigcup_{i \in \mathbb{N}} CCP^i$$

Theorem (Simulation)

For all $i \in \mathbb{N}$,

$$CCP^i \subseteq AC^i(UstCONN_2)$$

Definition (Translation from AC^i to CCP^i)

Input: *Description of a uniform familly of Boolean Circuits* $C = (C_n)$ *in* AC^i.

Output: *Description of a familly of Proof Circuits* $P = (P_n)$ *in* CCP^i *so that for all* k, *for all* $\overrightarrow{b}$, $P_k(\overrightarrow{b}) \rightarrow_{ev} b_j$ *iff* $C_k(\overrightarrow{b})$ *evaluates to* j.

Definition (Translation from AC^i to CCP^i)

Input: *Description of a uniform familly of Boolean Circuits $C = (C_n)$ in AC^i.*

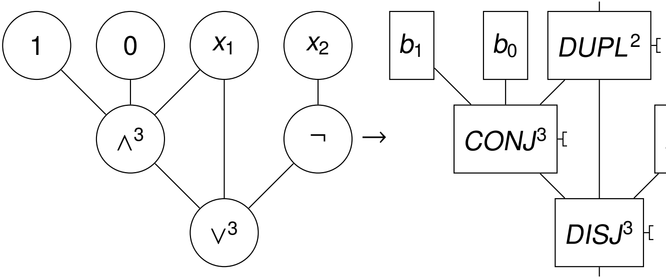
Output: *Description of a familly of Proof Circuits $P = (P_n)$ in CCP^i so that for all k, for all $\overrightarrow{b}$, $P_k(\overrightarrow{b}) \rightarrow_{ev} b_j$ iff $C_k(\overrightarrow{b})$ evaluates to j.*

Theorem

Translation from AC^i to CCP^i belongs to AC^0.

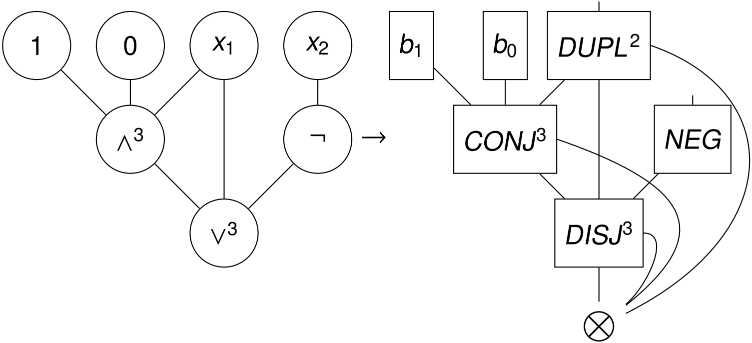
Theorem

Translation from AC^i to CCP^i belongs to AC^0.



Theorem

Translation from AC^i to CCP^i belongs to AC^0.



Lemmas ([Aubert, 2010])

- *For $n, m \in \mathbb{N}$, a gate of fan-in n and fan-out m labeled by a function of $\mathfrak{B}_1$ is translated with a piece of size $9(n+m)$.*

Lemmas ([Aubert, 2010])

- *For $n, m \in \mathbb{N}$, a gate of fan-in n and fan-out m labeled by a function of $\mathfrak{B}_1$ is translated with a piece of size $9(n+m)$.*
- *The depth of a piece and the arity of the function it represents are independent.*

Size and depth of Proof Circuits

Lemmas ([Aubert, 2010])

- *For $n, m \in \mathbb{N}$, a gate of fan-in n and fan-out m labeled by a function of $\mathfrak{B}_1$ is translated with a piece of size $9(n+m)$.*
- *The depth of a piece and the arity of the function it represents are independent.*

Theorem ([Aubert, 2010])

For all $i \in \mathbb{N}$, if $C = (C_n) \in AC^i$ (resp. NC^i), then its translation $P = (P_n)$ is such that for all $j \in \mathbb{N}$

- *the size of P_j is linear (resp. quadratic) in the size of C_j,*

Size and depth of Proof Circuits

Lemmas ([Aubert, 2010])

- *For $n, m \in \mathbb{N}$, a gate of fan-in n and fan-out m labeled by a function of $\mathfrak{B}_1$ is translated with a piece of size $9(n+m)$.*
- *The depth of a piece and the arity of the function it represents are independent.*

Theorem ([Aubert, 2010])

For all $i \in \mathbb{N}$, if $C = (C_n) \in AC^i$ (resp. NC^i), then its translation $P = (P_n)$ is such that for all $j \in \mathbb{N}$

- *the size of P_j is linear (resp. quadratic) in the size of C_j,*
- *the depth of P_j is linear in the depth of C_j.*

Corollaries

- *Uniformity is preserved*
- *Translation is lighter and takes NC into account*
- *For all* $i \in \mathbb{N}$, $AC^i \subseteq PCC^i \subseteq AC^i(UstConn_2)$

Corollaries

- *Uniformity is preserved*
- *Translation is lighter and takes NC into account*
- *For all $i \in \mathbb{N}$, $AC^i \subseteq PCC^i \subseteq AC^i(UstConn_2)$*

Future work:

- $UstConn_2 \in L$, is there a lighter function?
- Link between Proof Nets and Alternating Turing Machines?

Bibliography... and thanks

Aubert, C. (2010).
Réseaux de preuves booléens sous-logarithmiques.
Mémoire de M2 L.M.F.I., Paris VII, L.I.P.N.

Mogbil, V. and Rahli, V. (2007).
Uniform circuits, & Boolean proof nets.
In *Proceedings of LFCS'07*, pages 401–421. Springer.

Terui, K. (2004).
Proof Nets and Boolean Circuits.
In *Proceedings of LICS'04*, pages 182–191.

Thanks!