Entretien de suivi de thèse

LIPN – Université Sorbonne Paris Nord

Boris Eng

Contexte

De la logique linéaire à la géométrie de l'interaction

Correspondance de Curry-Howard. $A \Rightarrow B$ fonction de A vers B
$\longmapsto$ exécution = élimination des coupures.

Contexte

De la logique linéaire à la géométrie de l'interaction

Correspondance de Curry-Howard. $A \Rightarrow B$ fonction de A vers B
$\longmapsto$ exécution = élimination des coupures.

Logique linéaire (Girard). $A \Rightarrow B = !A \multimap B$ $\qquad A, B := X_i \mid X_i^\perp \mid A \otimes B \mid A \parr B$ (MLL).

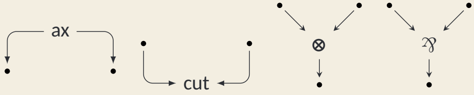
Contexte

De la logique linéaire à la géométrie de l'interaction

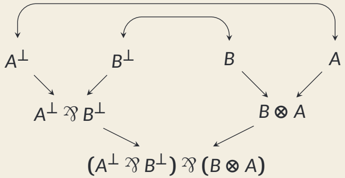
Correspondance de Curry-Howard. $A \Rightarrow B$ fonction de A vers B

$\longmapsto$ exécution = élimination des coupures.

Logique linéaire (Girard). $A \Rightarrow B = !A \multimap B$ $\qquad A, B := X_i \mid X_i^{\perp} \mid A \otimes B \mid A \parr B$ (MLL).



Axiome Coupure Tenseur Par



Contexte

De la Géométrie de l’Interaction à la Syntaxe Transcendantale

Géométrie de l’Interaction (Girard). S’abstraire des preuves. Mathématiser les réseaux.

Contexte

De la Géométrie de l'Interaction à la Syntaxe Transcendantale

Géométrie de l'Interaction (Girard). S'abstraire des preuves. Mathématiser les réseaux.

Syntaxe Transcendantale (Girard).

- successeur, 4 articles informels ;

Contexte

De la Géométrie de l'Interaction à la Syntaxe Transcendantale

Géométrie de l'Interaction (Girard). S'abstraire des preuves. Mathématiser les réseaux.

Syntaxe Transcendantale (Girard).

- successeur, 4 articles informels ;
- introduction d'un modèle de calcul (proche de la programmation logique) ;

Contexte

De la Géométrie de l’Interaction à la Syntaxe Transcendantale

Géométrie de l’Interaction (Girard). S’abstraire des preuves. Mathématiser les réseaux.

Syntaxe Transcendantale (Girard).

- successeur, 4 articles informels ;
- introduction d’un modèle de calcul (proche de la programmation logique) ;
- logique (linéaire) comme emergente du calcul.

Contexte

De la Géométrie de l'Interaction à la Syntaxe Transcendantale

Géométrie de l'Interaction (Girard). S'abstraire des preuves. Mathématiser les réseaux.

Syntaxe Transcendantale (Girard).

- successeur, 4 articles informels ;
- introduction d'un modèle de calcul (proche de la programmation logique) ;
- logique (linéaire) comme emergente du calcul.

$\longrightarrow$ Formalisation (pour MLL) avec *Thomas Seiller*.

Contexte

De la Géométrie de l'Interaction à la Syntaxe Transcendantale

Géométrie de l'Interaction (Girard). S'abstraire des preuves. Mathématiser les réseaux.

Syntaxe Transcendantale (Girard).

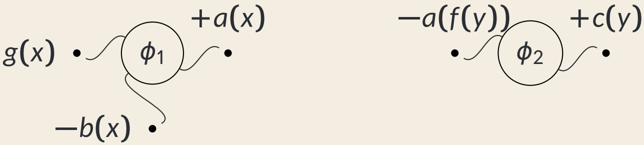
- successeur, 4 articles informels ;
- introduction d'un modèle de calcul (proche de la programmation logique) ;
- logique (linéaire) comme emergente du calcul.

$\longrightarrow$ Formalisation (pour MLL) avec *Thomas Seiller*.

$\longrightarrow$ Usage de la GdI pour la complexité en espace du λ-calcul avec *Damiano Mazza*.

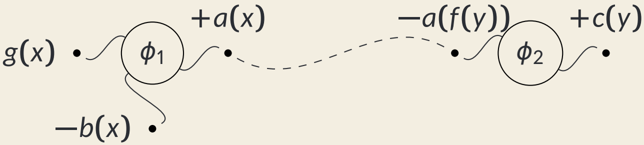
Résolution stellaire (stage M2 + 1ère année)

Formalisation naïve des étoiles et constellations



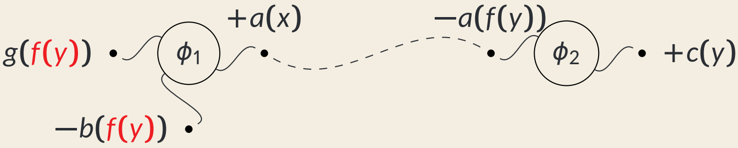
Résolution stellaire (stage M2 + 1ère année)

Formalisation naïve des étoiles et constellations



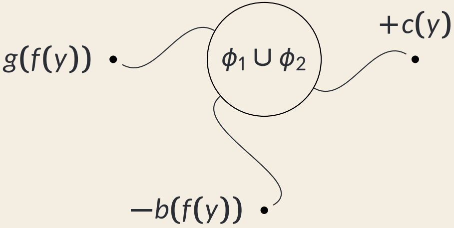
Résolution stellaire (stage M2 + 1ère année)

Formalisation naïve des étoiles et constellations



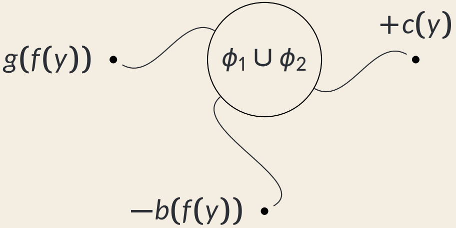
Résolution stellaire (stage M2 + 1ère année)

Formalisation naïve des étoiles et constellations



Résolution stellaire (stage M2 + 1ère année)

Formalisation naïve des étoiles et constellations



Constellation Φ (n étoiles)

$\downarrow$

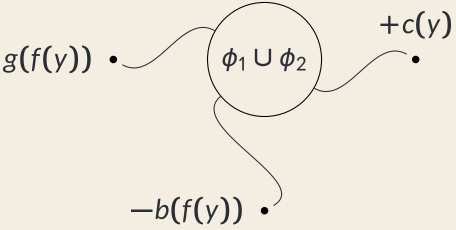
Diagrammes (connexions)

$\downarrow$

Constellation $\mathrm{Ex}(\Phi)$

Résolution stellaire (stage M2 + 1ère année)

Formalisation naïve des étoiles et constellations



Constellation Φ (n étoiles)

$\downarrow$

Diagrammes (connexions)

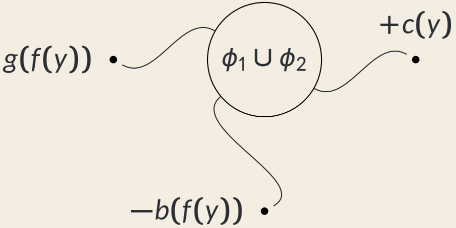
$\downarrow$

Constellation $\mathrm{Ex}(\Phi)$

Développement du modèle $\longleftrightarrow$ Simulation de la logique $\longleftrightarrow$ Exemples de calcul.

Résolution stellaire (stage M2 + 1ère année)

Formalisation naïve des étoiles et constellations



Constellation Φ (n étoiles)

$\downarrow$

Diagrammes (connexions)

$\downarrow$

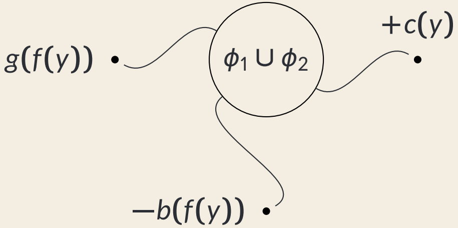
Constellation $\mathrm{Ex}(\Phi)$

Développement du modèle $\longleftrightarrow$ Simulation de la logique $\longleftrightarrow$ Exemples de calcul.

1. Liens (conjecturés) : programmation logique, tuiles de Wang, pavages biologiques.

Résolution stellaire (stage M2 + 1ère année)

Formalisation naïve des étoiles et constellations



Constellation Φ (n étoiles)

$\downarrow$

Diagrammes (connexions)

$\downarrow$

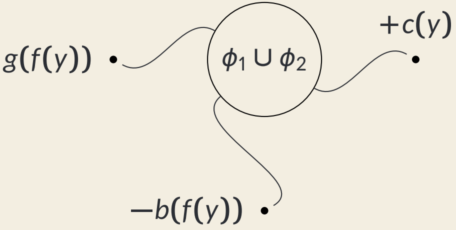
Constellation $\mathrm{Ex}(\Phi)$

Développement du modèle $\longleftrightarrow$ Simulation de la logique $\longleftrightarrow$ Exemples de calcul.

1. Liens (conjecturés) : programmation logique, tuiles de Wang, pavages biologiques.
   $[g(x), -b(x), \underline{+a(x)}] + [\underline{-a(f(y))}, +c(y)] \longrightarrow [g(f(y)), -b(f(y)), +c(y)].$

Résolution stellaire (stage M2 + 1ère année)

Formalisation naïve des étoiles et constellations



Constellation Φ (n étoiles)

$\downarrow$

Diagrammes (connexions)

$\downarrow$

Constellation $\mathrm{Ex}(\Phi)$

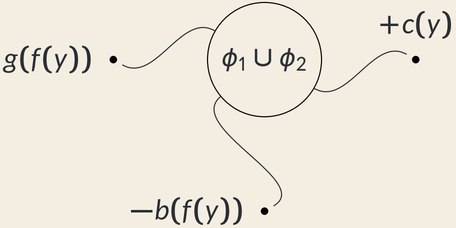
Développement du modèle $\longleftrightarrow$ Simulation de la logique $\longleftrightarrow$ Exemples de calcul.

1. Liens (conjecturés) : programmation logique, tuiles de Wang, pavages biologiques.

$$[g(x), -b(x), \underline{+a(x)}] + [\underline{-a(f(y))}, +c(y)] \longrightarrow [g(f(y)), -b(f(y)), +c(y)].$$

Résolution stellaire (stage M2 + 1ère année)

Formalisation naïve des étoiles et constellations

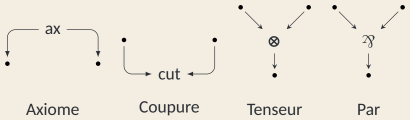


$$\begin{array}{c}\text{Constellation } \Phi \text{ (}n\text{ étoiles)}\\ \downarrow \\ \text{Diagrammes (connexions)} \\ \downarrow \\ \text{Constellation Ex}(\Phi)\end{array}$$

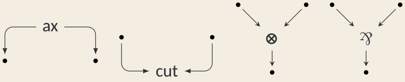
Développement du modèle $\longleftrightarrow$ Simulation de la logique $\longleftrightarrow$ Exemples de calcul.

1. Liens (conjecturés) : programmation logique, tuiles de Wang, pavages biologiques.
   $$[g(x), -b(x), \underline{+a(x)}] + [\underline{-a(f(y))}, +c(y)] \longrightarrow [g(f(y)), -b(f(y)), +c(y)].$$
2. Vérification de propriétés essentielles (confluence, terminaison, ...).

Reconstruction de MLL (1ère année)



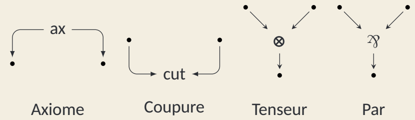
Reconstruction de MLL (1ère année)



Axiome Coupure Tenseur Par

Double nature, preuve = calcul + logique.

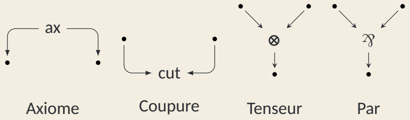
Reconstruction de MLL (1ère année)



Double nature, preuve = calcul + logique.

Calcul. Encoder les structures de preuves $\mathscr{S}$ de MLL (hyperarête = étoile).

Reconstruction de MLL (1ère année)

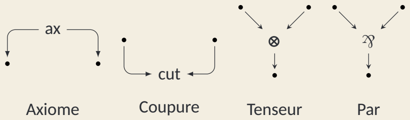


Double nature, preuve = calcul + logique.

Calcul. Encoder les structures de preuves $\mathscr{S}$ de MLL (hyperarête = étoile).

Logique. $\mathscr{S} \rightsquigarrow \Phi_{\mathscr{S}}$. Est-ce que $\Phi_{\mathscr{S}}$ est une vraie preuve ?

Reconstruction de MLL (1ère année)



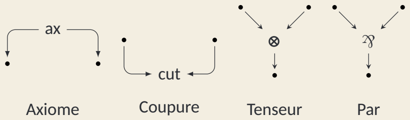
Double nature, preuve = calcul + logique.

Calcul. Encoder les structures de preuves $\mathscr{S}$ de MLL (hyperarête = étoile).

Logique. $\mathscr{S} \rightsquigarrow \Phi_{\mathscr{S}}$. Est-ce que $\Phi_{\mathscr{S}}$ est une vraie preuve ?

- Critère de Danos-Regnier : faire passer des tests $\Phi_1, \ldots, \Phi_n$.

Reconstruction de MLL (1ère année)



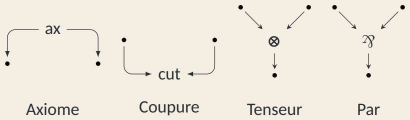
Double nature, preuve = calcul + logique.

Calcul. Encoder les structures de preuves $\mathscr{S}$ de MLL (hyperarête = étoile).

Logique. $\mathscr{S} \rightsquigarrow \Phi_{\mathscr{S}}$. Est-ce que $\Phi_{\mathscr{S}}$ est une vraie preuve ?

- Critère de Danos-Regnier : faire passer des tests $\Phi_1, \ldots, \Phi_n$.
- $\mathrm{Ex}(\Phi_{\mathscr{S}} \uplus \Phi_i)$ satisfait une propriété $P \longrightarrow$ orthogonalité $\Phi_{\mathscr{S}} \perp \Phi_i$.

Reconstruction de MLL (1ère année)



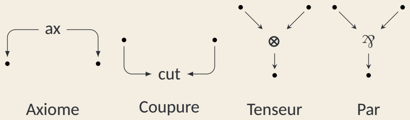
Double nature, preuve = calcul + logique.

Calcul. Encoder les structures de preuves $\mathscr{S}$ de MLL (hyperarête = étoile).

Logique. $\mathscr{S} \rightsquigarrow \Phi_{\mathscr{S}}$. Est-ce que $\Phi_{\mathscr{S}}$ est une vraie preuve ?

- Critère de Danos-Regnier : faire passer des tests $\Phi_1, \ldots, \Phi_n$.
- $\mathrm{Ex}(\Phi_{\mathscr{S}} \uplus \Phi_i)$ satisfait une propriété $P \longrightarrow$ orthogonalité $\Phi_{\mathscr{S}} \perp \Phi_i$.
- Formules par réalisabilité $\mathbf{A} = \{\Phi_1, \Phi_2, \ldots\}$, $\mathbf{A}^{\perp}$, $\mathbf{A} \otimes \mathbf{B}$, $\mathbf{A} \parr \mathbf{B}$.

Reconstruction de MLL (1ère année)



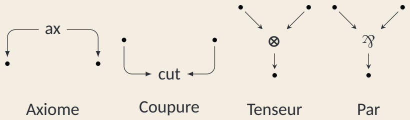
Double nature, preuve = calcul + logique.

Calcul. Encoder les structures de preuves $\mathcal{S}$ de MLL (hyperarête = étoile).

Logique. $\mathcal{S} \rightsquigarrow \Phi_{\mathcal{S}}$. Est-ce que $\Phi_{\mathcal{S}}$ est une vraie preuve ?

- Critère de Danos-Regnier : faire passer des tests $\Phi_1, \ldots, \Phi_n$.
- $\mathrm{Ex}(\Phi_{\mathcal{S}} \uplus \Phi_i)$ satisfait une propriété $P \longrightarrow$ orthogonalité $\Phi_{\mathcal{S}} \perp \Phi_i$.
- Formules par réalisabilité $\mathbf{A} = \{\Phi_1, \Phi_2, \ldots\}$, $\mathbf{A}^\perp$, $\mathbf{A} \otimes \mathbf{B}$, $\mathbf{A} \parr \mathbf{B}$.
- Correction et complétude : on simule/capture bien MLL (et MLL+MIX).

Reconstruction de MLL (1ère année)



Double nature, preuve = calcul + logique.

Calcul. Encoder les structures de preuves $\mathscr{S}$ de MLL (hyperarête = étoile).

Logique. $\mathscr{S} \rightsquigarrow \Phi_{\mathscr{S}}$. Est-ce que $\Phi_{\mathscr{S}}$ est une vraie preuve ?

- Critère de Danos-Regnier : faire passer des tests $\Phi_1, \ldots, \Phi_n$.
- $\mathrm{Ex}(\Phi_{\mathscr{S}} \uplus \Phi_i)$ satisfait une propriété $P \longrightarrow$ orthogonalité $\Phi_{\mathscr{S}} \perp \Phi_i$.
- Formules par réalisabilité $\mathbf{A} = \{\Phi_1, \Phi_2, \ldots\}$, $\mathbf{A}^{\perp}$, $\mathbf{A} \otimes \mathbf{B}$, $\mathbf{A} \parr \mathbf{B}$.
- Correction et complétude : on simule/capture bien MLL (et MLL+MIX).

Soumission d'article (rejetée). "Stellar Resolution : Multiplicatives", CSL 2020 (Été).

Exploration en largeur et rédaction (début 2ème année)

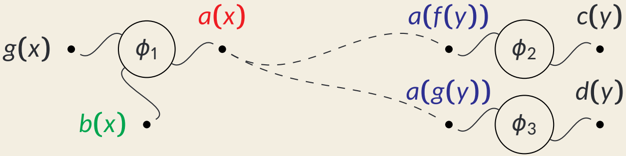
- Notes personnelles :

Exploration en largeur et rédaction (début 2ème année)

- Notes personnelles :
  - machines à compteur, automates cellulaires / à pile, machines de Turing ;

Exploration en largeur et rédaction (début 2ème année)

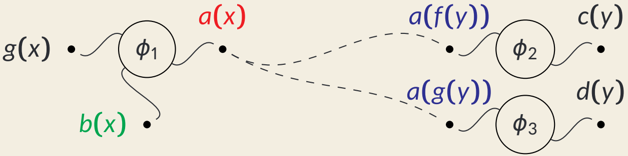
- Notes personnelles :
  - machines à compteur, automates cellulaires / à pile, machines de Turing ;
  - extensions informelles vers MALL : non-déterminisme, connexions interdites



[Abandon] ;

Exploration en largeur et rédaction (début 2ème année)

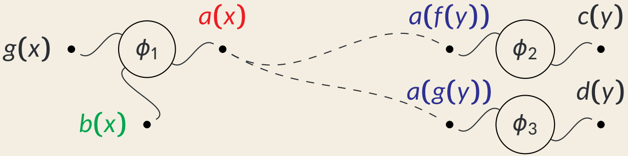
- Notes personnelles :
  - machines à compteur, automates cellulaires / à pile, machines de Turing ;
  - extensions informelles vers MALL : non-déterminisme, connexions interdites



    [Abandon] ;
  - extensions informelles vers MELL : gérer la duplication/l'effacement en logique ;

Exploration en largeur et rédaction (début 2ème année)

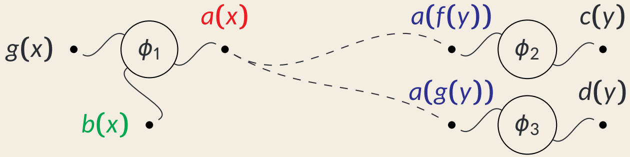
- Notes personnelles :
  - machines à compteur, automates cellulaires / à pile, machines de Turing ;
  - extensions informelles vers MALL : non-déterminisme, connexions interdites



    [Abandon] ;
  - extensions informelles vers MELL : gérer la duplication/l'effacement en logique ;
- Amélioration/développement des définitions et preuves de l'article CSL2020.

Exploration en largeur et rédaction (début 2ème année)

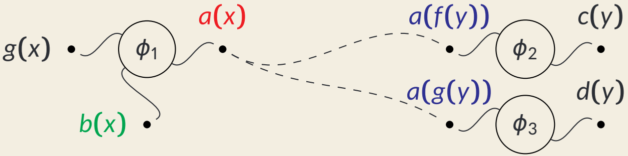
- Notes personnelles :
  - machines à compteur, automates cellulaires / à pile, machines de Turing ;
  - extensions informelles vers MALL : non-déterminisme, connexions interdites



    [Abandon] ;
  - extensions informelles vers MELL : gérer la duplication/l'effacement en logique ;
- Amélioration/développement des définitions et preuves de l'article CSL2020.
- Preuve de simulation des "modèles d'assemblage de tuiles abstraites" (bioinformatique).

Exploration en largeur et rédaction (début 2ème année)

- Notes personnelles :
  - machines à compteur, automates cellulaires / à pile, machines de Turing ;
  - extensions informelles vers MALL : non-déterminisme, connexions interdites

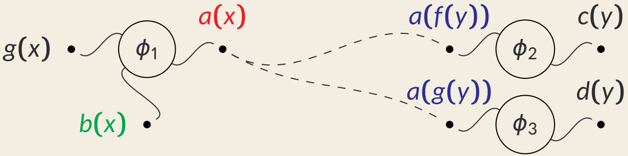


    [Abandon] ;
  - extensions informelles vers MELL : gérer la duplication/l'effacement en logique ;
- Amélioration/développement des définitions et preuves de l'article CSL2020.
- Preuve de simulation des "modèles d'assemblage de tuiles abstraites" (bioinformatique).

Systèmes de types atypiques ?

Exploration en largeur et rédaction (début 2ème année)

- Notes personnelles :
  - machines à compteur, automates cellulaires / à pile, machines de Turing ;
  - extensions informelles vers MALL : non-déterminisme, connexions interdites

    [Abandon] ;
  - extensions informelles vers MELL : gérer la duplication/l'effacement en logique ;
- Amélioration/développement des définitions et preuves de l'article CSL2020.
- Preuve de simulation des "modèles d'assemblage de tuiles abstraites" (bioinformatique).

Systèmes de types atypiques ?

Soumission d'article (rejetée). LICS 2020 (Fin automne).

Travail avec Damiano Mazza

Lambda-calcul et complexité en espace

Thèse de Church-Turing. Machines de Turing et Lambda-calcul $\longmapsto$ même "puissance".

Travail avec Damiano Mazza

Lambda-calcul et complexité en espace

Thèse de Church-Turing. Machines de Turing et Lambda-calcul $\longmapsto$ même "puissance".

Thèse d'invariance. Même "efficacité" ? Comment mesurer l'espace du λ-calcul ?

Travail avec Damiano Mazza

Lambda-calcul et complexité en espace

Thèse de Church-Turing. Machines de Turing et Lambda-calcul $\longmapsto$ même "puissance".

Thèse d'invariance. Même "efficacité" ? Comment mesurer l'espace du λ-calcul ?

$\longmapsto$ Géométrie de l'interaction $=$ exploration statique de λ-terme (Schöpp).

Travail avec Damiano Mazza

Lambda-calcul et complexité en espace

Thèse de Church-Turing. Machines de Turing et Lambda-calcul $\longmapsto$ même "puissance".

Thèse d'invariance. Même "efficacité" ? Comment mesurer l'espace du λ-calcul ?

$\longmapsto$ Géométrie de l'interaction $=$ exploration statique de λ-terme (Schöpp).

$\longmapsto$ relation entre $\lambda\mathtt{Space}$ et $\mathtt{Space}(O(f))$ (stage Master 1).

Travail avec Damiano Mazza

Lambda-calcul et complexité en espace

Thèse de Church-Turing. Machines de Turing et Lambda-calcul $\longmapsto$ même "puissance".

Thèse d'invariance. Même "efficacité" ? Comment mesurer l'espace du λ-calcul ?

$\longmapsto$ Géométrie de l'interaction $=$ exploration statique de λ-terme (Schöpp).

$\longmapsto$ relation entre $\lambda\mathsf{Space}$ et $\mathsf{Space}(O(f))$ (stage Master 1).

Durant la thèse :

Travail avec Damiano Mazza

Lambda-calcul et complexité en espace

Thèse de Church-Turing. Machines de Turing et Lambda-calcul $\longmapsto$ même "puissance".

Thèse d'invariance. Même "efficacité" ? Comment mesurer l'espace du λ-calcul ?

$\longmapsto$ Géométrie de l'interaction $=$ exploration statique de λ-terme (Schöpp).

$\longmapsto$ relation entre $\lambda\mathtt{Space}$ et $\mathtt{Space}(O(f))$ (stage Master 1).

Durant la thèse :

- exportation vers PCF ou le λ-calcul CBPV.

Travail avec Damiano Mazza

Lambda-calcul et complexité en espace

Thèse de Church-Turing. Machines de Turing et Lambda-calcul $\longmapsto$ même "puissance".

Thèse d'invariance. Même "efficacité" ? Comment mesurer l'espace du λ-calcul ?

$\longmapsto$ Géométrie de l'interaction = exploration statique de λ-terme (Schöpp).

$\longmapsto$ relation entre $\lambda\mathsf{Space}$ et $\mathsf{Space}(O(f))$ (stage Master 1).

Durant la thèse :

- exportation vers PCF ou le λ-calcul CBPV.
- outils catégoriques pour approximation du calcul : proto-(op)fibrations.

Travail avec Damiano Mazza

Lambda-calcul et complexité en espace

Thèse de Church-Turing. Machines de Turing et Lambda-calcul $\longmapsto$ même "puissance".

Thèse d'invariance. Même "efficacité" ? Comment mesurer l'espace du λ-calcul ?

$\longmapsto$ Géométrie de l'interaction $=$ exploration statique de λ-terme (Schöpp).

$\longmapsto$ relation entre $\lambda\mathtt{Space}$ et $\mathtt{Space}(O(f))$ (stage Master 1).

Durant la thèse :

- exportation vers PCF ou le λ-calcul CBPV.
- outils catégoriques pour approximation du calcul : proto-(op)fibrations.
- "Interacting Seems Unreasonable, in Time and Space" (Accattoli, Dal Lago, Vanoni).

Travail avec Damiano Mazza

Lambda-calcul et complexité en espace

Thèse de Church-Turing. Machines de Turing et Lambda-calcul $\mapsto$ même "puissance".

Thèse d'invariance. Même "efficacité" ? Comment mesurer l'espace du λ-calcul ?

$\mapsto$ Géométrie de l'interaction = exploration statique de λ-terme (Schöpp).

$\mapsto$ relation entre $\lambda\mathtt{Space}$ et $\mathtt{Space}(O(f))$ (stage Master 1).

Durant la thèse :

- exportation vers PCF ou le λ-calcul CBPV.
- outils catégoriques pour approximation du calcul : proto-(op)fibrations.
- "Interacting Seems Unreasonable, in Time and Space" (Accattoli, Dal Lago, Vanoni).

Résolution stellaire $\mapsto$ MLL $\mapsto$ MELL $\mapsto$ λ-calcul. Pas d'intérêt évident...

Rédaction journal (Printemps 2ème année)

Rédaction d'article. LMCS – Logical Methods in Computer Science.

Rédaction journal (Printemps 2ème année)

Rédaction d'article. LMCS – Logical Methods in Computer Science.

Rédaction journal (Printemps 2ème année)

Rédaction d'article. LMCS – Logical Methods in Computer Science.

- Détails de toutes les preuves, preuves de simulations :

Rédaction journal (Printemps 2ème année)

Rédaction d'article. LMCS – Logical Methods in Computer Science.

- Détails de toutes les preuves, preuves de simulations :
  - automates, machines de Turing, circuits booléens/arithmétiques ;

Rédaction journal (Printemps 2ème année)

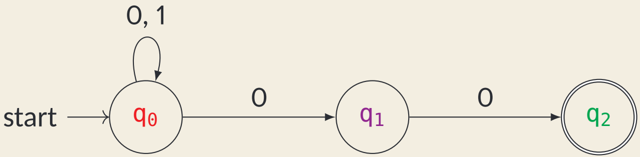
Rédaction d'article. LMCS – Logical Methods in Computer Science.

- Détails de toutes les preuves, preuves de simulations :
  - automates, machines de Turing, circuits booléens/arithmétiques ;
  - modèle d'assemblage de tuiles abstraites.

Rédaction journal (Printemps 2ème année)

Rédaction d'article. LMCS – Logical Methods in Computer Science.

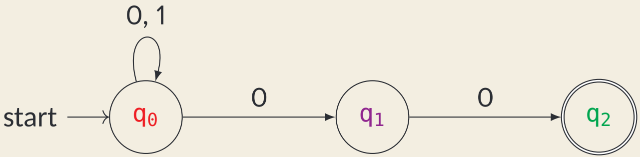
- Détails de toutes les preuves, preuves de simulations :
  - automates, machines de Turing, circuits booléens/arithmétiques ;
  - modèle d'assemblage de tuiles abstraites.
- Résolution stellaire : transfert de données local dans un réseau (hypergraphe).



Rédaction journal (Printemps 2ème année)

Rédaction d'article. LMCS – Logical Methods in Computer Science.

- Détails de toutes les preuves, preuves de simulations :
  - automates, machines de Turing, circuits booléens/arithmétiques ;
  - modèle d'assemblage de tuiles abstraites.
- Résolution stellaire : transfert de données local dans un réseau (hypergraphe).

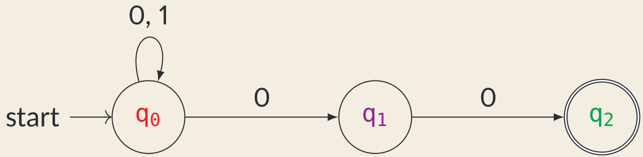


$[-i(w), +a(w, q_0)]+$

Rédaction journal (Printemps 2ème année)

Rédaction d'article. LMCS – Logical Methods in Computer Science.

- Détails de toutes les preuves, preuves de simulations :
  - automates, machines de Turing, circuits booléens/arithmétiques ;
  - modèle d'assemblage de tuiles abstraites.
- Résolution stellaire : transfert de données local dans un réseau (hypergraphe).

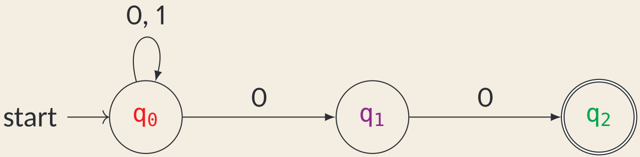


$$[-i(w), +a(w, q_0)] + [-a(0 \cdot w, q_0), +a(w, q_0)] + [-a(1 \cdot w, q_0), +a(w, q_0)] +$$

Rédaction journal (Printemps 2ème année)

Rédaction d'article. LMCS – Logical Methods in Computer Science.

- Détails de toutes les preuves, preuves de simulations :
  - automates, machines de Turing, circuits booléens/arithmétiques ;
  - modèle d'assemblage de tuiles abstraites.
- Résolution stellaire : transfert de données local dans un réseau (hypergraphe).

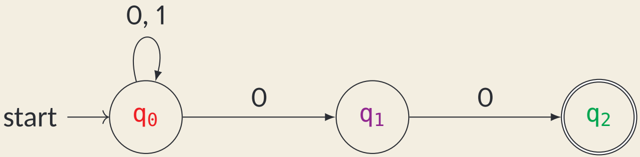


$$[-i(w), +a(w, q_0)] + [-a(0 \cdot w, q_0), +a(w, q_0)] + [-a(1 \cdot w, q_0), +a(w, q_0)] +$$
$$[-a(0 \cdot w, q_0), +a(w, q_1)] +$$

Rédaction journal (Printemps 2ème année)

Rédaction d'article. LMCS – Logical Methods in Computer Science.

- Détails de toutes les preuves, preuves de simulations :
  - automates, machines de Turing, circuits booléens/arithmétiques ;
  - modèle d'assemblage de tuiles abstraites.
- Résolution stellaire : transfert de données local dans un réseau (hypergraphe).

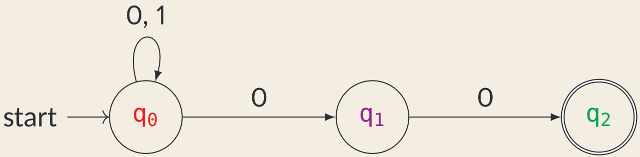


$$[-i(w), +a(w, \mathtt{q_0})] + [-a(0 \cdot w, \mathtt{q_0}), +a(w, \mathtt{q_0})] + [-a(1 \cdot w, \mathtt{q_0}), +a(w, \mathtt{q_0})] +$$
$$[-a(0 \cdot w, \mathtt{q_0}), +a(w, \mathtt{q_1})] + [-a(0 \cdot w, \mathtt{q_1}), +a(w, \mathtt{q_2})] +$$

Rédaction journal (Printemps 2ème année)

Rédaction d'article. LMCS – Logical Methods in Computer Science.

- Détails de toutes les preuves, preuves de simulations :
  - automates, machines de Turing, circuits booléens/arithmétiques ;
  - modèle d'assemblage de tuiles abstraites.
- Résolution stellaire : transfert de données local dans un réseau (hypergraphe).



$$[-i(w), +a(w, \mathtt{q_0})] + [-a(0 \cdot w, \mathtt{q_0}), +a(w, \mathtt{q_0})] + [-a(1 \cdot w, \mathtt{q_0}), +a(w, \mathtt{q_0})] +$$
$$[-a(0 \cdot w, \mathtt{q_0}), +a(w, \mathtt{q_1})] + [-a(0 \cdot w, \mathtt{q_1}), +a(w, \mathtt{q_2})] + [-a(\epsilon, \mathtt{q_2}), \mathtt{accept}]$$

Rédaction journal (Été 2ème année)

Deux notions de types *unifiées* :

Rédaction journal (Été 2ème année)

Deux notions de types *unifiées* :

- **types primitifs** (programmation $f : A$, calcul des séquents $\vdash \pi : A$, ...) ;

Rédaction journal (Été 2ème année)

Deux notions de types *unifiées* :

- **types primitifs** (programmation $f : A$, calcul des séquents $\vdash \pi : A$, ...) ;
- **types par réalisabilité** $t \Vdash \mathbf{A}$ (description comportementale) avec $\mathbf{A} = \{t_1, t_2, \dots\}$;

Rédaction journal (Été 2ème année)

Deux notions de types *unifiées* :

- **types primitifs** (programmation $f : A$, calcul des séquents $\vdash \pi : A$, ...) ;
- **types par réalisabilité** $t \Vdash \mathbf{A}$ (description comportementale) avec $\mathbf{A} = \{t_1, t_2, \dots\}$;
- reliés par *adéquation* : si $\Phi : A$ alors $\Phi \Vdash \mathbf{A}$;

Rédaction journal (Été 2ème année)

Deux notions de types *unifiées* :

- **types primitifs** (programmation $f : A$, calcul des séquents $\vdash \pi : A$, ...) ;
- **types par réalisabilité** $t \Vdash \mathbf{A}$ (description comportementale) avec $\mathbf{A} = \{t_1, t_2, \dots\}$;
- reliés par *adéquation* : si $\Phi : A$ alors $\Phi \Vdash \mathbf{A}$;
- interprétation en syntaxe transcendantale.

Rédaction journal (Été 2ème année)

Deux notions de types *unifiées* :

- **types primitifs** (programmation $f : A$, calcul des séquents $\vdash \pi : A$, ...) ;
- **types par réalisabilité** $t \Vdash \mathbf{A}$ (description comportementale) avec $\mathbf{A} = \{t_1, t_2, \dots\}$;
- reliés par *adéquation* : si $\Phi : A$ alors $\Phi \Vdash \mathbf{A}$;
- interprétation en syntaxe transcendantale.

Extensions possibles

Rédaction journal (Été 2ème année)

Deux notions de types *unifiées* :

- **types primitifs** (programmation $f : A$, calcul des séquents $\vdash \pi : A$, ...) ;
- **types par réalisabilité** $t \Vdash \mathbf{A}$ (description comportementale) avec $\mathbf{A} = \{t_1, t_2, \dots\}$;
- reliés par *adéquation* : si $\Phi : A$ alors $\Phi \Vdash \mathbf{A}$;
- interprétation en syntaxe transcendantale.

Extensions possibles

- transposer les résultats à MELL (déjà un peu exploré) $\longmapsto$ λ-calcul.

Rédaction journal (Été 2ème année)

Deux notions de types *unifiées* :

- **types primitifs** (programmation $f : A$, calcul des séquents $\vdash \pi : A$, ...) ;
- **types par réalisabilité** $t \Vdash \mathbf{A}$ (description comportementale) avec $\mathbf{A} = \{t_1, t_2, \dots\}$;
- reliés par *adéquation* : si $\Phi : A$ alors $\Phi \Vdash \mathbf{A}$;
- interprétation en syntaxe transcendantale.

Extensions possibles

- transposer les résultats à MELL (déjà un peu exploré) $\longmapsto$ λ-calcul.
- étendre à la logique du second ordre $\longmapsto$ System F.

Rédaction journal (Été 2ème année)

Deux notions de types *unifiées* :

- **types primitifs** (programmation $f : A$, calcul des séquents $\vdash \pi : A$, ...) ;
- **types par réalisabilité** $t \Vdash \mathbf{A}$ (description comportementale) avec $\mathbf{A} = \{t_1, t_2, \dots\}$;
- reliés par *adéquation* : si $\Phi : A$ alors $\Phi \Vdash \mathbf{A}$;
- interprétation en syntaxe transcendantale.

Extensions possibles

- transposer les résultats à MELL (déjà un peu exploré) $\longmapsto$ λ-calcul.
- étendre à la logique du second ordre $\longmapsto$ System F.
- étendre à la logique du premier ordre $\longmapsto$ complexité descriptive ?

Rédaction journal (Été 2ème année)

Deux notions de types *unifiées* :

- **types primitifs** (programmation $f : A$, calcul des séquents $\vdash \pi : A$, ...) ;
- **types par réalisabilité** $t \Vdash \mathbf{A}$ (description comportementale) avec $\mathbf{A} = \{t_1, t_2, \ldots\}$;
- reliés par *adéquation* : si $\Phi : A$ alors $\Phi \Vdash \mathbf{A}$;
- interprétation en syntaxe transcendantale.

Extensions possibles

- transposer les résultats à MELL (déjà un peu exploré) $\longmapsto$ λ-calcul.
- étendre à la logique du second ordre $\longmapsto$ System F.
- étendre à la logique du premier ordre $\longmapsto$ complexité descriptive ?
- application à l'étude des classes de complexité.

Merci d'avoir écouté ma présentation.