Tâches, Types et Tactiques pour les Systèmes de Calculs Locaux

Pierre Castéran, Vincent Filou
JFLA10

janvier 2010

Introduction

Preuve formelles de système de calculs locaux en *coq* :

- Certification d'algorithmes
- Méta-théorie

Faire le lien entre des spécifications globales
et des implémentations par *interactions locales*

Théorie des calculs locaux

Calculs Locaux

- Y. Métivier, E.Sopena : Graph Relabelling Systems : A General Overview [1997]
- J. Chalopin, Y. Métivier : A Bridge Between the Asynchronous Message Passing Model and Local Computations in Graphs [2005]
- J. Chalopin, E. Godard, Y. Métivier : Local Terminations and Distributed Computability in Anonymous Networks [2008]
- Y. Métivier, J.-M. Robson, N. Saheb-Djahromi, and A. Zemmari. An optimal bit complexity randomised distributed MIS algorithm [2009]

Le modèle des calculs locaux

Réseau

On représente un réseau par un graphe non orienté, simple et connexe

- Chaque sommet représente un processeur.
- Chaque arête représente un lien de communication directe.

État

Un étiquetage du graphe symbolise l'état du réseau. On notera $\Sigma_{G,\mathcal{L}}$ le type des états du graphe G ou les étiquettes sont de type $\mathcal{L}$.

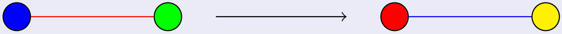
Calculs locaux

Une règle de calcul local est une relation de réétiquetage dont la portée est réduite à une boule de rayon 1, commutant par isomorphisme de graphes étiquetés.

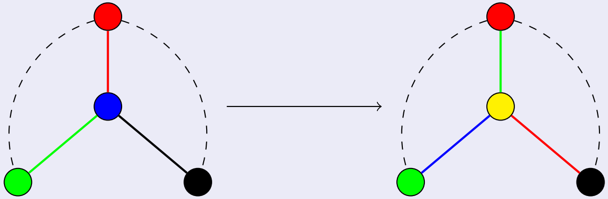
Un pas de calcul local.

Classes de systèmes de calculs locaux

$LC0$

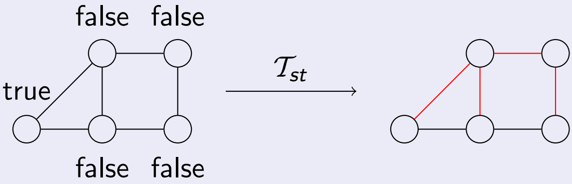


$LC1$

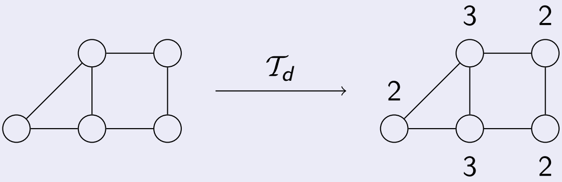


Notion de tâche

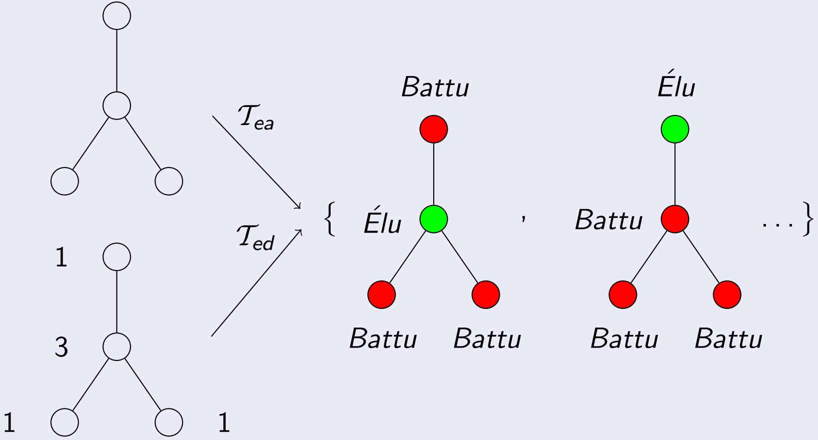
$\mathcal{T}_{st}$: Calcul d'arbre recouvrant



$\mathcal{T}_d$: calcul du degré de chaque sommet



$\mathcal{T}_{ea}$ et $\mathcal{T}_{ed}$: Élections dans un arbre

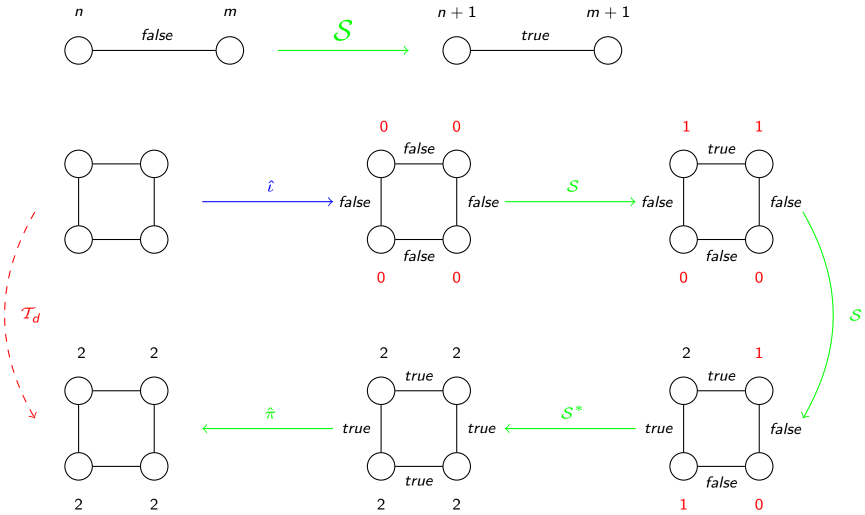


Réalisation ITD d'une tâche par un système de réétiquetage

Soit $\mathcal{S}$ un système noetherien

$$\begin{tikzcd}
i : \Sigma_{G,\mathcal{L}} \arrow[rrr, "\mathcal{S}^*"] & & & f : \Sigma_{G,\mathcal{L}} \not\longrightarrow \arrow[d, "\hat{\pi}"'] \\
I : \Sigma_{G,\mathcal{L}_i} \arrow[u, "\hat{\iota}"] \arrow[rrr, dashed, "\mathcal{T}"] & & & F : \Sigma_{G,\mathcal{L}_O}
\end{tikzcd}$$

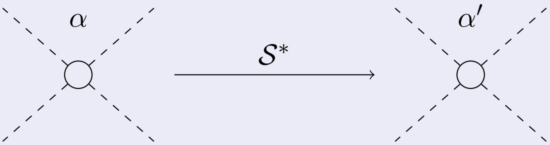
Une réalisation $LC0$ de $\mathcal{T}_d$ (calcul du degré)



Detection de la terminaison

Detection Locale de la Terminaison (*LTD*)

- Un sommet "sait" quand *sa* valeur utile (π) calculée est finale.
- Quand le calcul est terminé, tout sommet sait que *sa* valeur utile est finale.



$$\begin{array}{c} \tau(\alpha) = \mathit{true} \\ \pi(\alpha) = v \end{array} \quad \Rightarrow \quad \begin{array}{c} \tau(\alpha') = \mathit{true} \\ \pi(\alpha') = \pi(\alpha) = v \end{array}$$

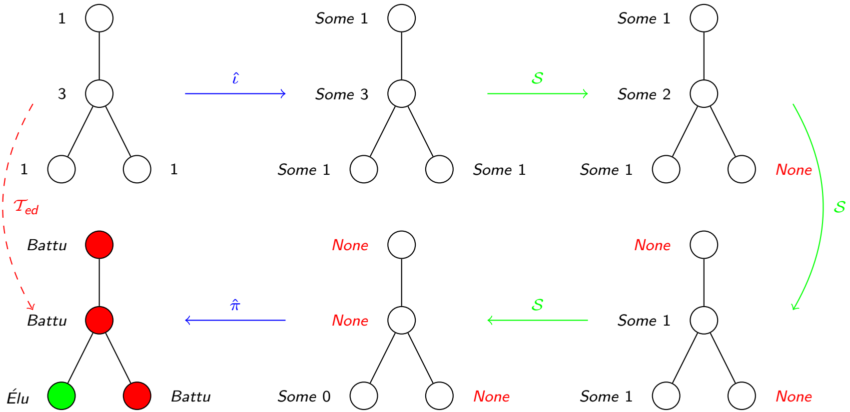
Une réalisation $LC0 - LTD$ de $\mathcal{T}_{ed}$ (1)

Élection dans un arbre, avec connaissance initiale du degré

$$\begin{aligned}
\tau(Some\ 0) &= true \\
\tau(None) &= true \\
\tau(Some\ (S\ n)) &= false
\end{aligned}$$

$$\begin{aligned}
\pi(Some\ 0) &= \acute{E}lu \\
\pi(None) &= Battu \\
\pi(Some\ n) &= quelconque
\end{aligned}$$

Une réalisation $LC0$ de $\mathcal{T}_{ed}$ (2)



Preuve de correction

Preuve par récurrence sur les séquences de réétiquetage.

Invariant

- Le sous-graphe des sommets marqués par *Some n* est un arbre
- Pour tout sommet marqué *Some n*, *n* est égale au degré de ce sommet dans ledit sous graphe.

Terminaison

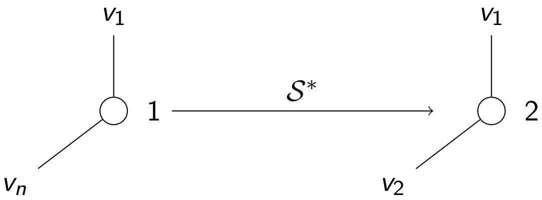
- Ordre : $\forall n, None < Some\ n$
- $\{\{Some\ (n+1), Some\ 1\}\} \ll \{\{Some\ n, None\}\}$

États initiaux, mode de terminaison et expressivité

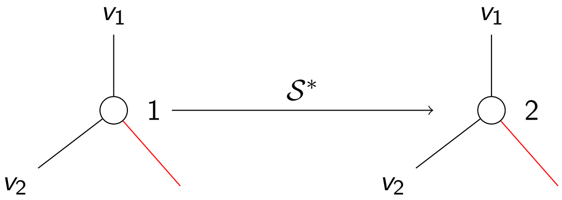
Tâche	**Classe**	**états initiaux**	**Terminaison**	**Réalisation**
Election	LC0	degré	LTD	Oui
Election	LC0	uniforme	LTD	Impossible
Degré	LC0	uniforme	ITD	Oui
Degré	LC1	uniforme	LTD	Oui
Degré	LC0	uniforme	LTD	Impossible

Preuve d'impossibilité : calcul du degré

On suppose l'existence d'un algorithme $LC0$, LTD, réalisant la tâche $\mathcal{T}_d$:



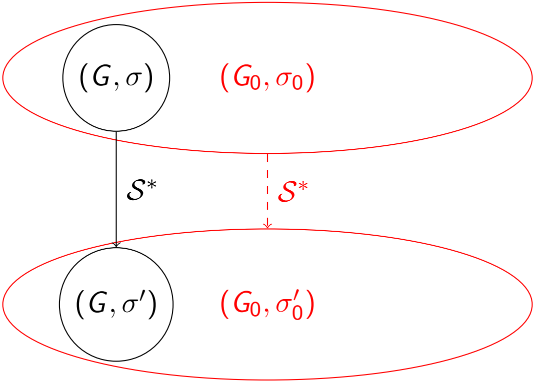
$\tau(\alpha') = true$
$\pi(\alpha') = 2$



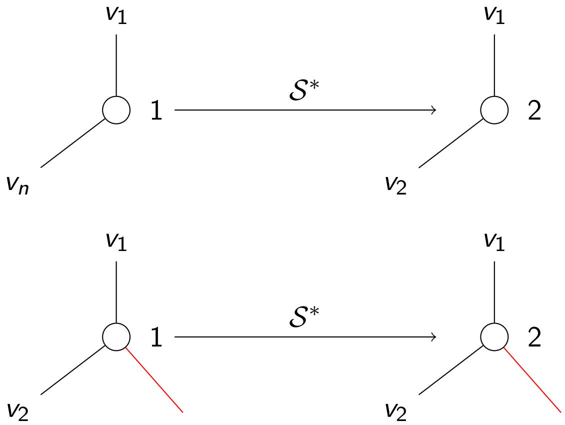
$\tau(\alpha') = true$
$\pi(\alpha') = 3$

Lemme d'extension

Pour tout type d'étiquetage L, tout système $LC0$ $\mathcal{S}$, tout graphe G et tout état σ ...



Preuve d'impossibilité : calcul du degré



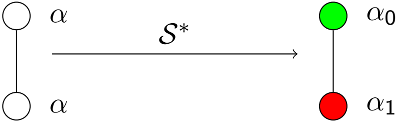
$\tau(\alpha') = \mathit{true}$
$\pi(\alpha') = 2$

$\tau(\alpha') = \mathit{true}$
$\pi(\alpha') = 2$

$$2 = 3$$

Élection dans un graphe uniforme

On considère un algorithme $\mathcal{S}$ réalisant la tâche $\mathcal{T}_{ea}$:



$\tau(\alpha_0) = \textit{true}$

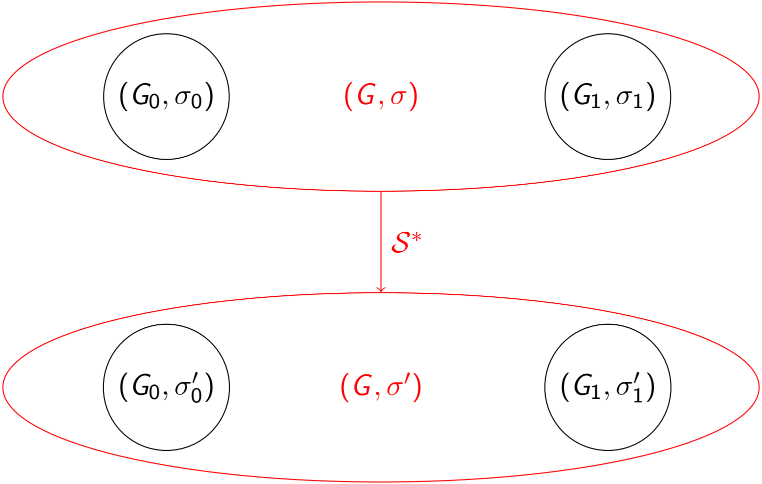
$\pi(\alpha_0) =$Élu.

Lemme de composition (1)

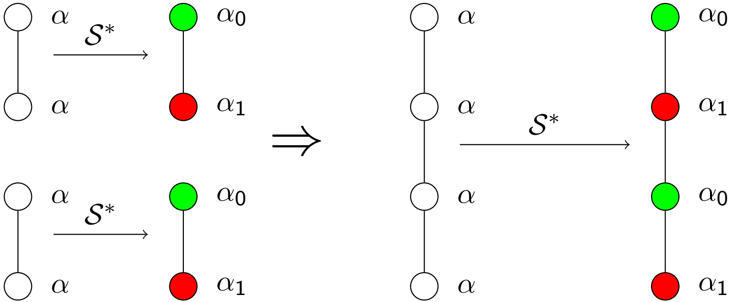
Pour tout type d'étiquetage L, tout système $LC0$ $\mathcal{S}$, tout couple de graphes disjoints G_0 et G_1, ...

$$
\begin{tikzcd}
(G_0, \sigma_0) \arrow[d, "\mathcal{S}^*"] & & (G_1, \sigma_1) \arrow[d, "\mathcal{S}^*"] \\
(G_0, \sigma'_0) & & (G_1, \sigma'_1)
\end{tikzcd}
$$

Lemme de composition (2)



Composition et élection



$\tau(\alpha_0) = \textit{true}$

$\pi(\alpha_0) =$Élu.

Conclusion

Résultats

- Certification des algorithmes “classiques” (élection, arbre recouvrant)
- Les spécifications et algorithmes sont des “citoyens de première classe”.
- Preuves d'impossibilités
- Tout algorithme $LC0$ est un calcul Local.
- Preuve d'inclusion de classes d'algorithmes
  - inclusion de modes de terminaison (transformations certifiés d'algorithmes)
- Passage d'un langage semi-formel à un langage formel

Conclusion

Travaux futurs

- Classe $LC2$, Outils théoriques $LC1$ et $LC2$
- Amélioration des structures de données utilisées.
- Plus d'automatisation, langage de preuve plus lisible.
- Compositions d'algorithmes.
- Algorithmes probabilistes.
- Graphes dynamiques.