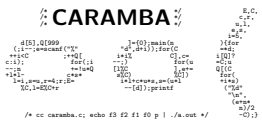
Paramétrage de CADO-NFS pour le problème du logarithme discret dans les corps finis premiers

Kevin Trancho
M1 Informatique

Université Paris-Est Marne-la-Vallée
Stage encadré par Pierrick Gaudry

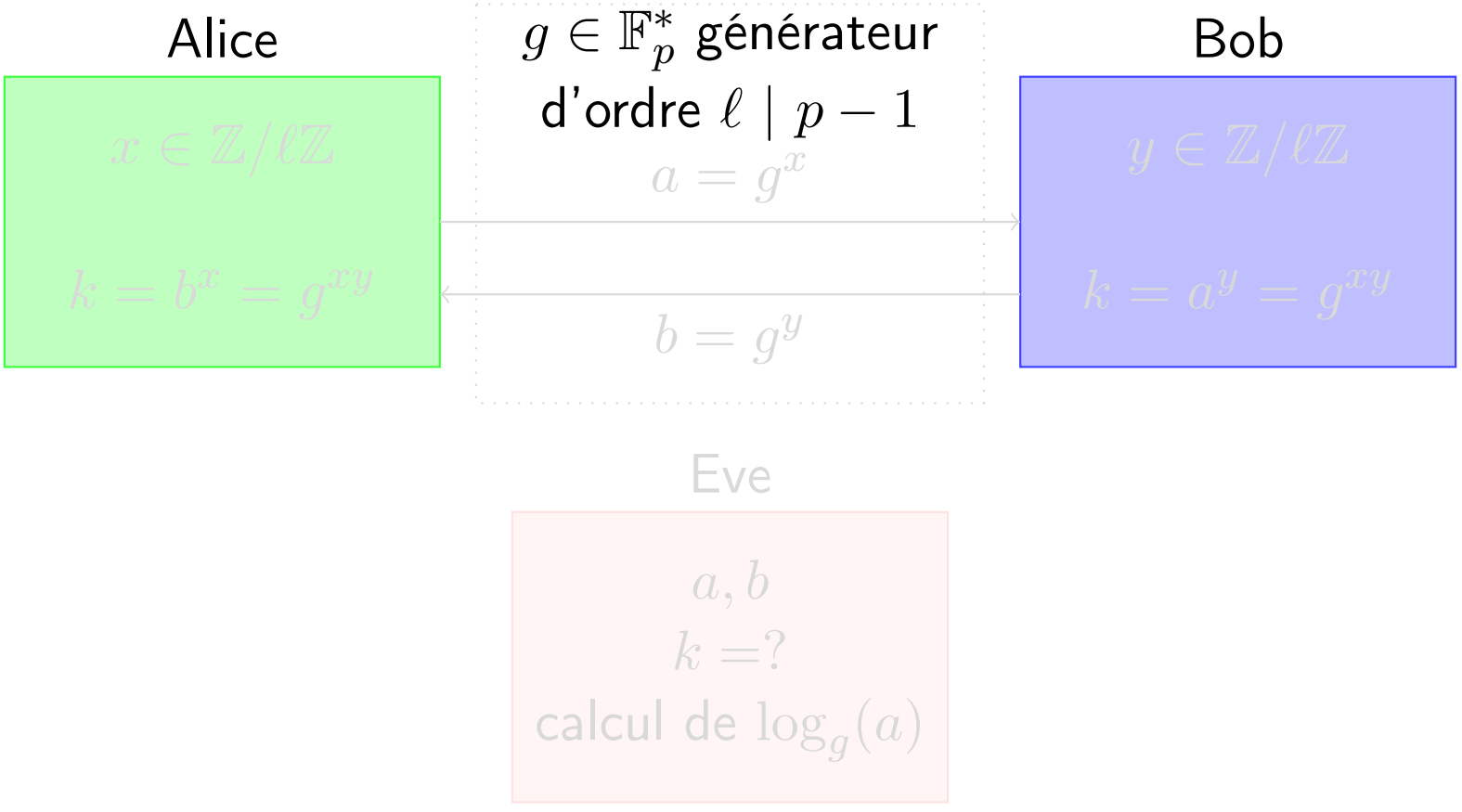
31 août 2018



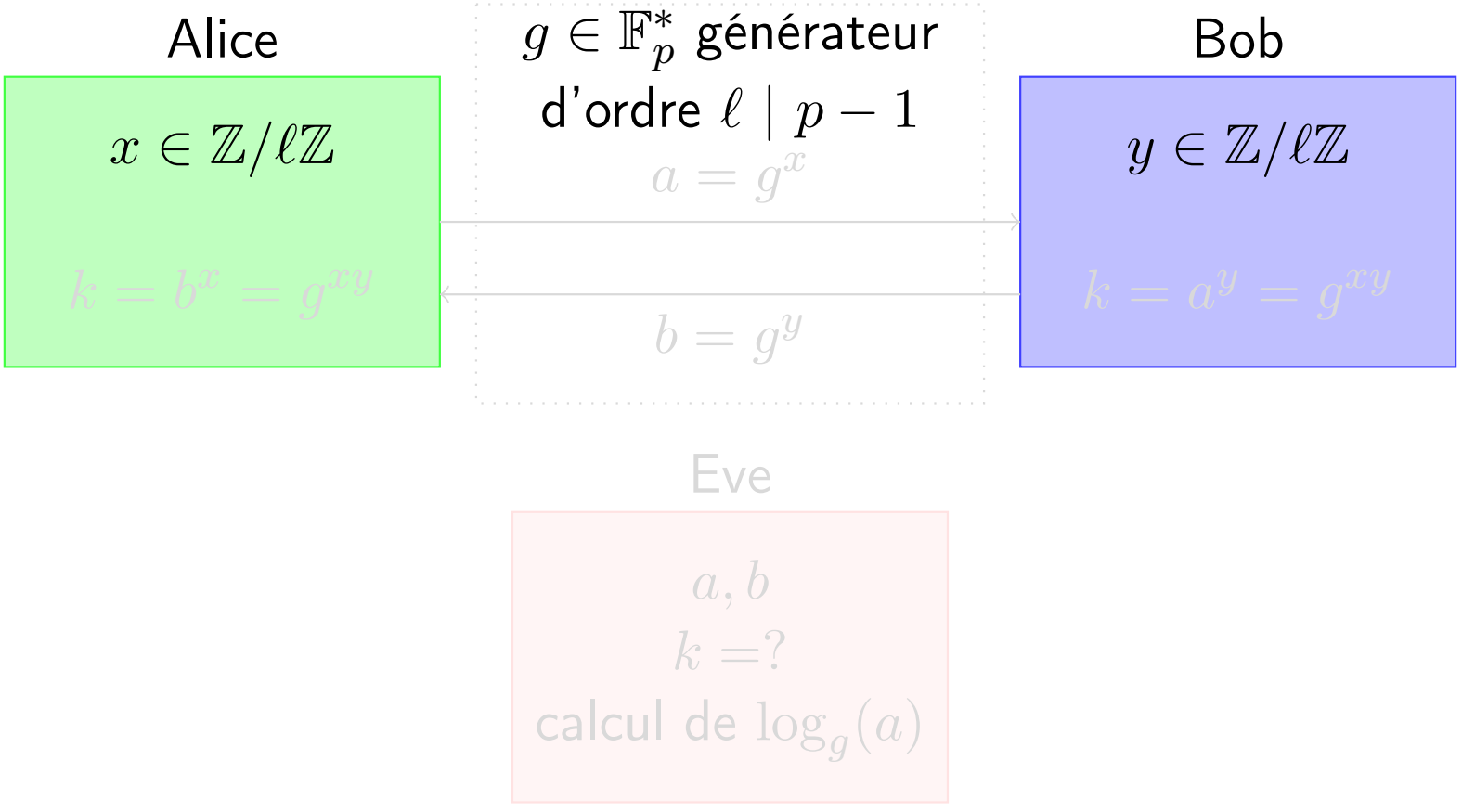
Au programme de l'exposé

- Paramétrage de CADO-NFS, un problème difficile.
- Différences de comportement entre les petites et grandes tailles.
- La sélection polynomiale Joux-Lercier : une alternative intéressante à l'algorithme de Kleinjung pour le logarithme discret pour les petites tailles ?

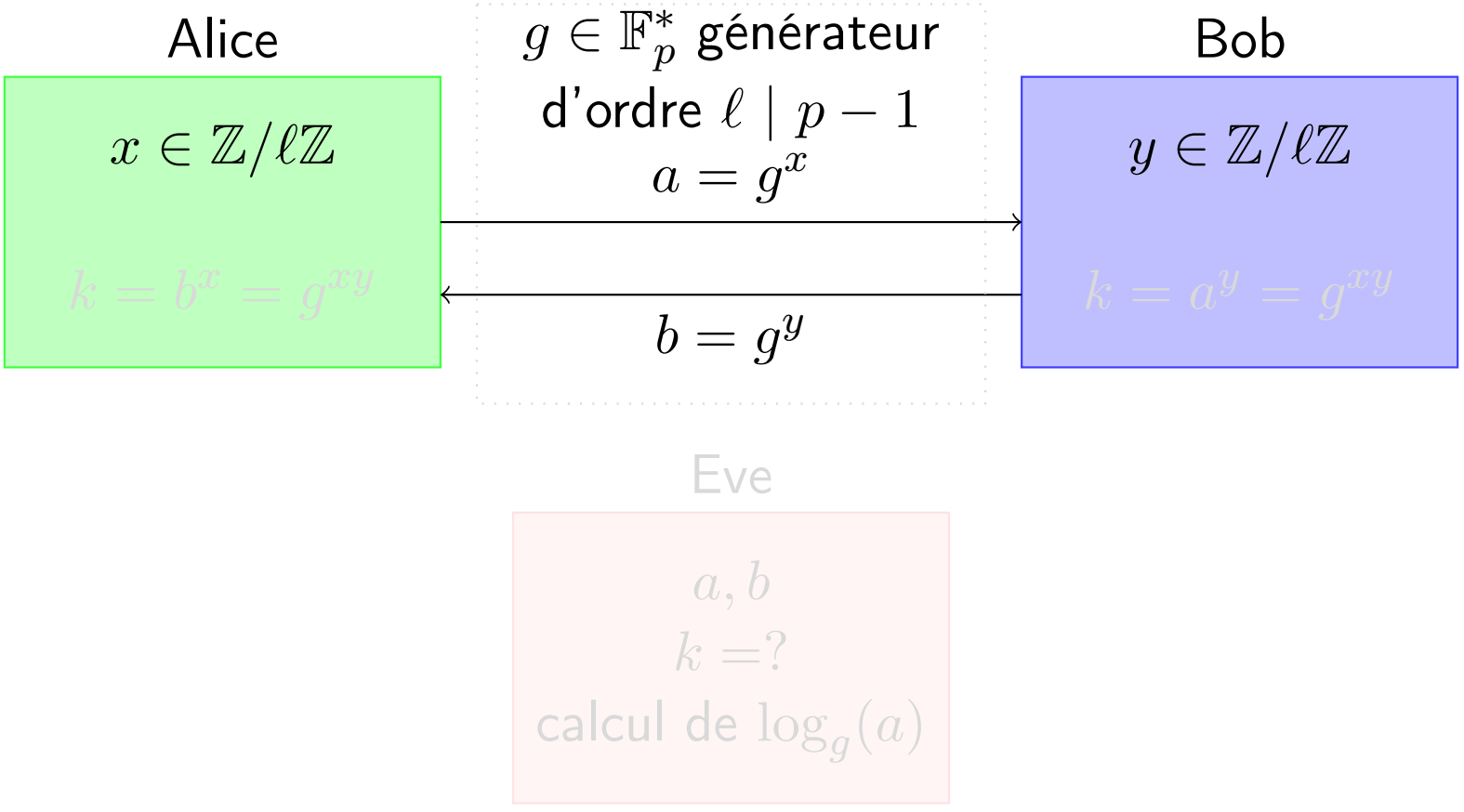
Un exemple en pratique : l'échange de clés de Diffie-Hellman



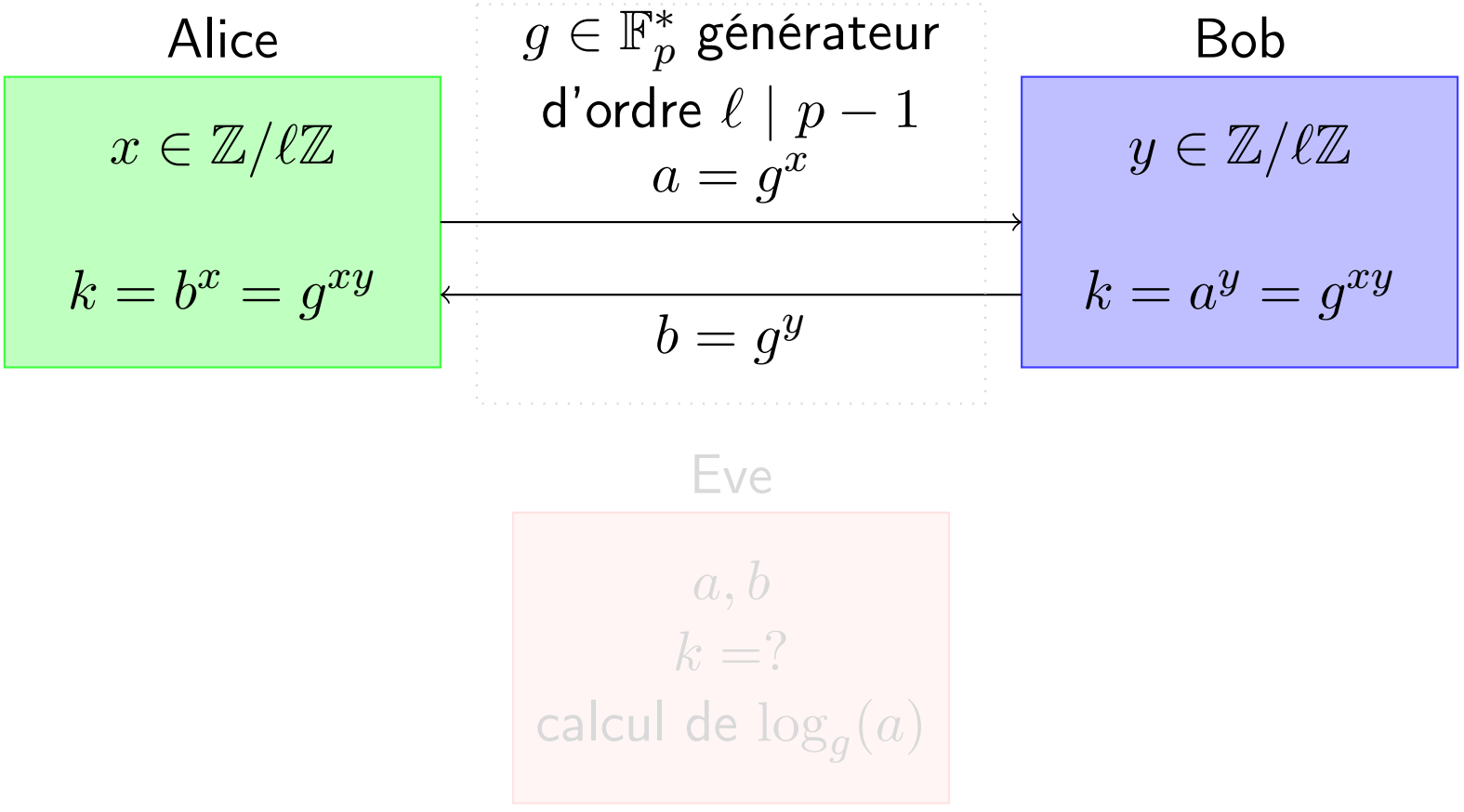
Un exemple en pratique : l'échange de clés de Diffie-Hellman



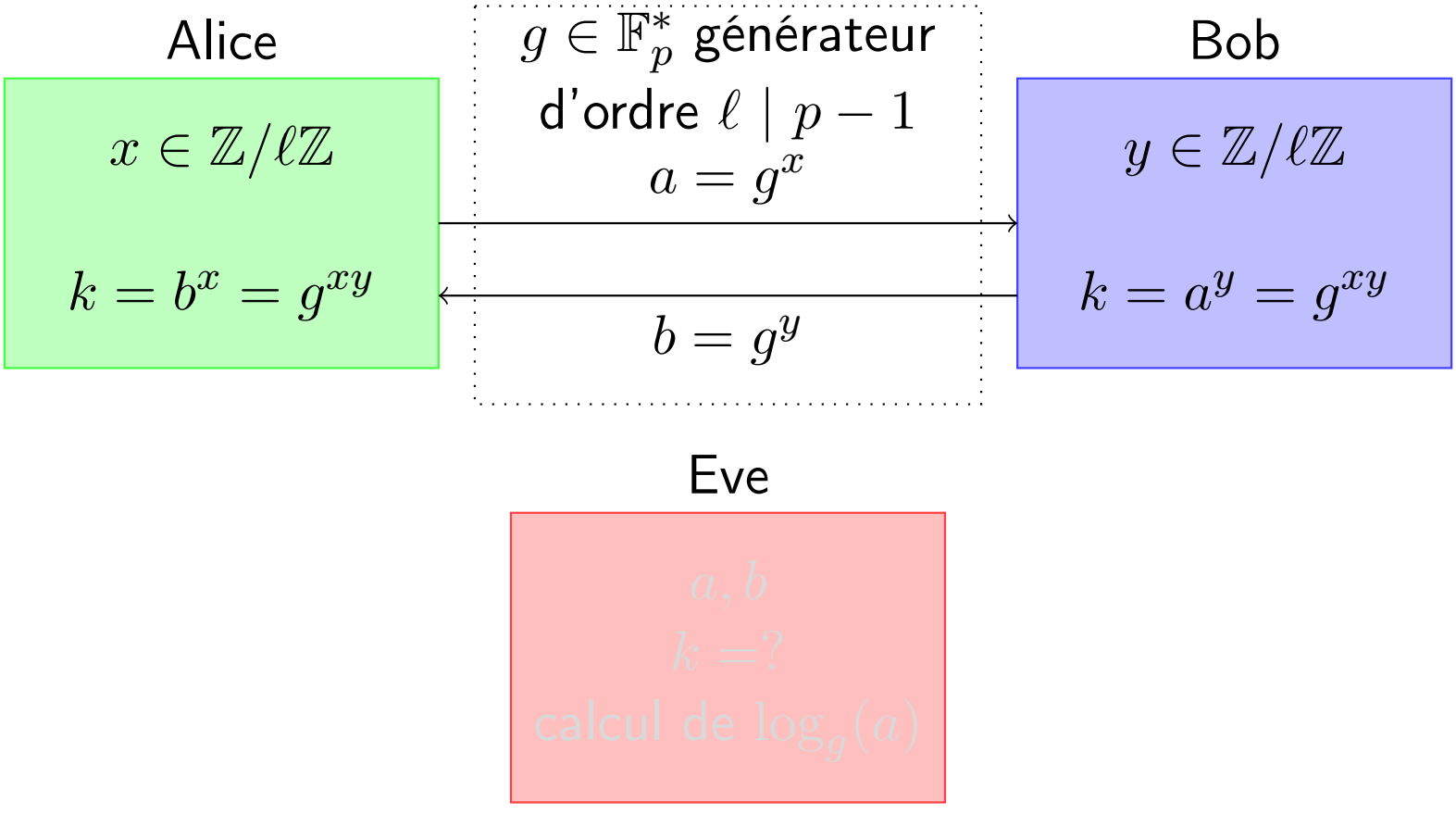
Un exemple en pratique : l'échange de clés de Diffie-Hellman



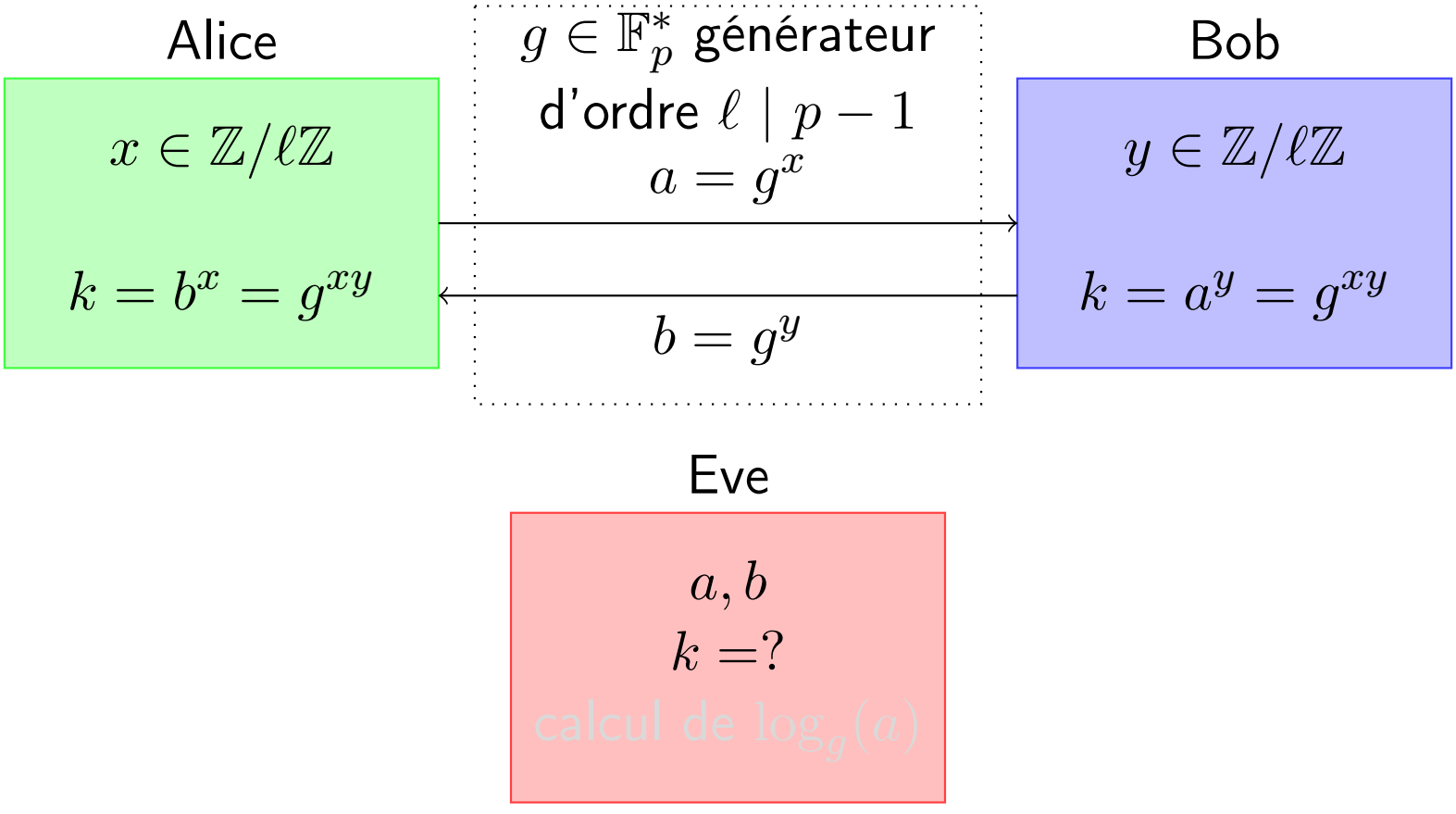
Un exemple en pratique : l'échange de clés de Diffie-Hellman



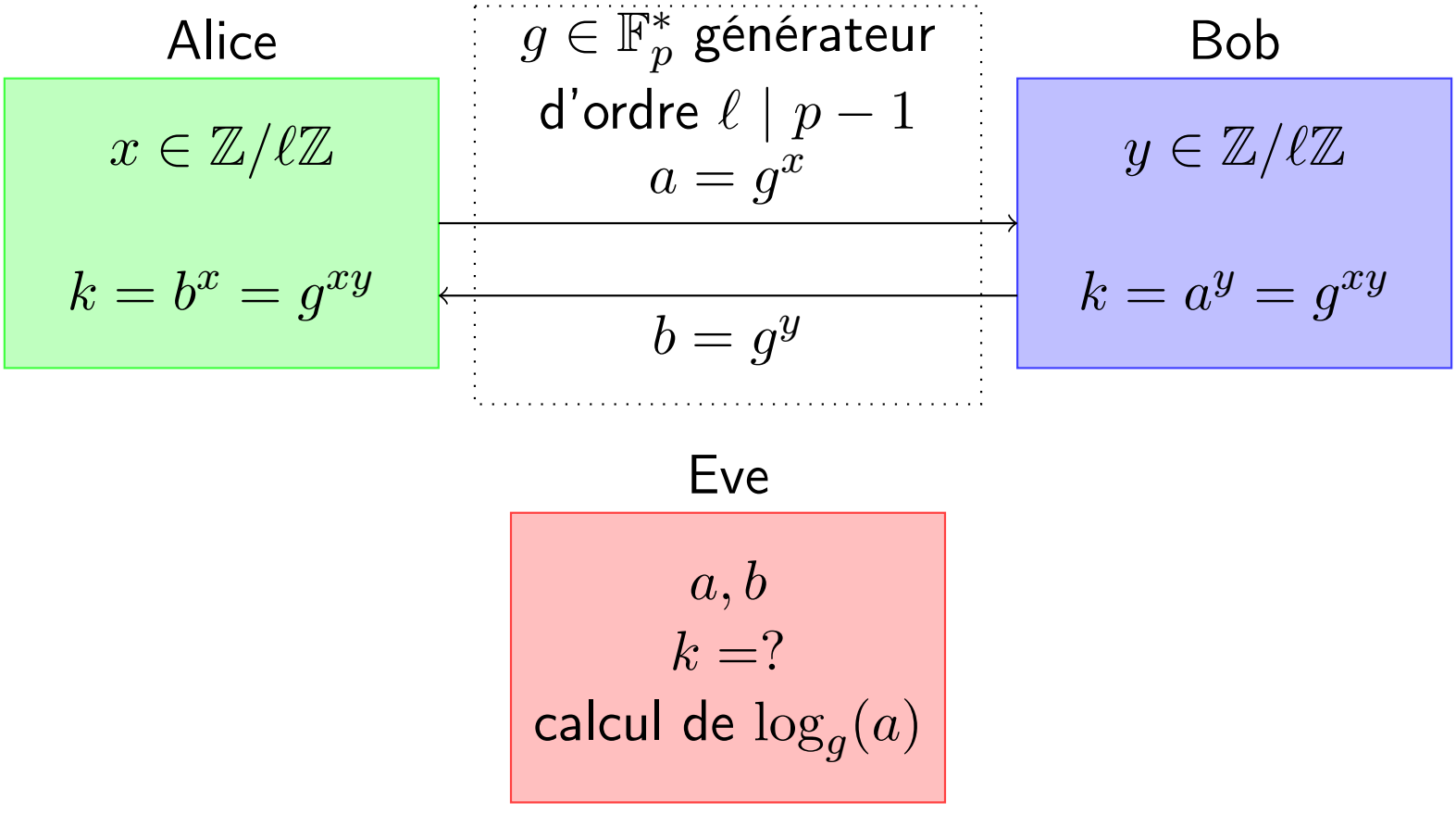
Un exemple en pratique : l'échange de clés de Diffie-Hellman



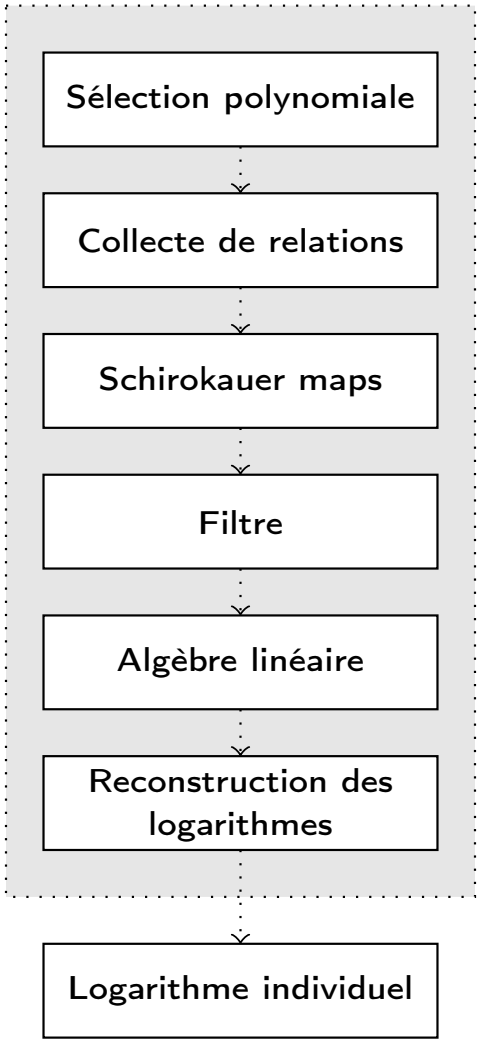
Un exemple en pratique : l'échange de clés de Diffie-Hellman



Un exemple en pratique : l'échange de clés de Diffie-Hellman



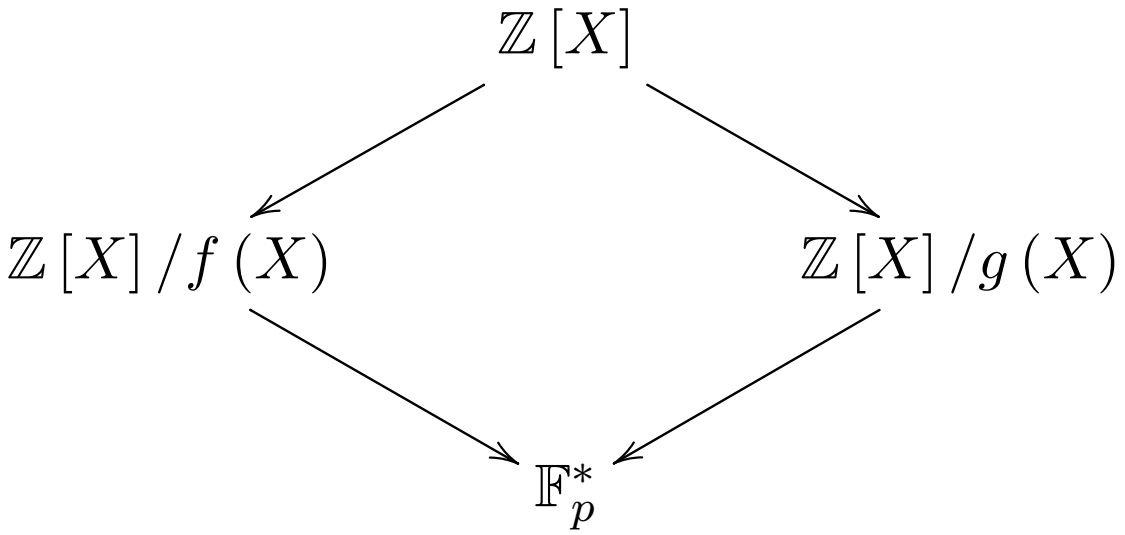
CADO-NFS : grandes étapes



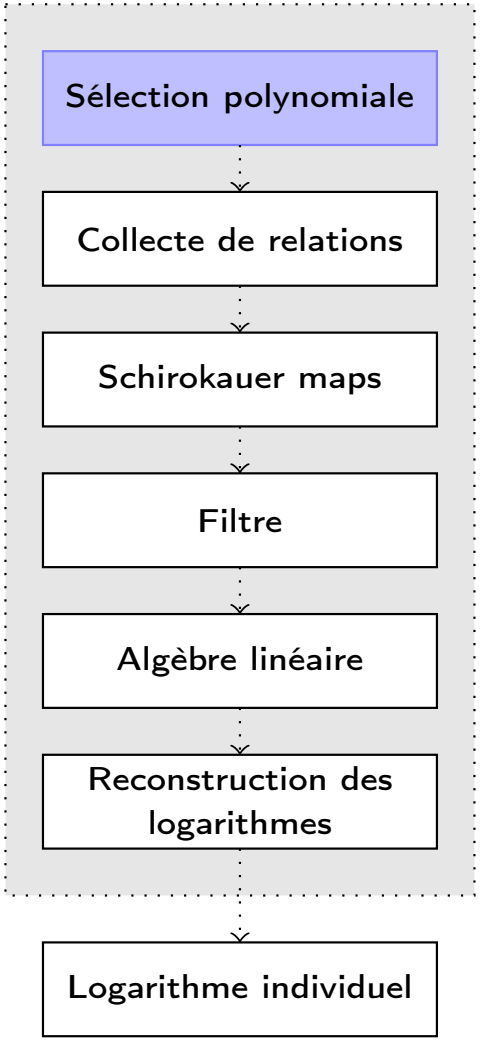
CADO-NFS : Sélection polynomiale

On cherche deux polynômes f et g tels que :

- f et g sont irréductibles dans $\mathbb{Z}[X]$.
- f et g ont une racine commune modulo p.



Égalité dans $\mathbb{F}_p^*$.



CADO-NFS : Sélection polynomiale

À un paire $(a, b) \in \mathbb{Z}^2$, on associe les normes :

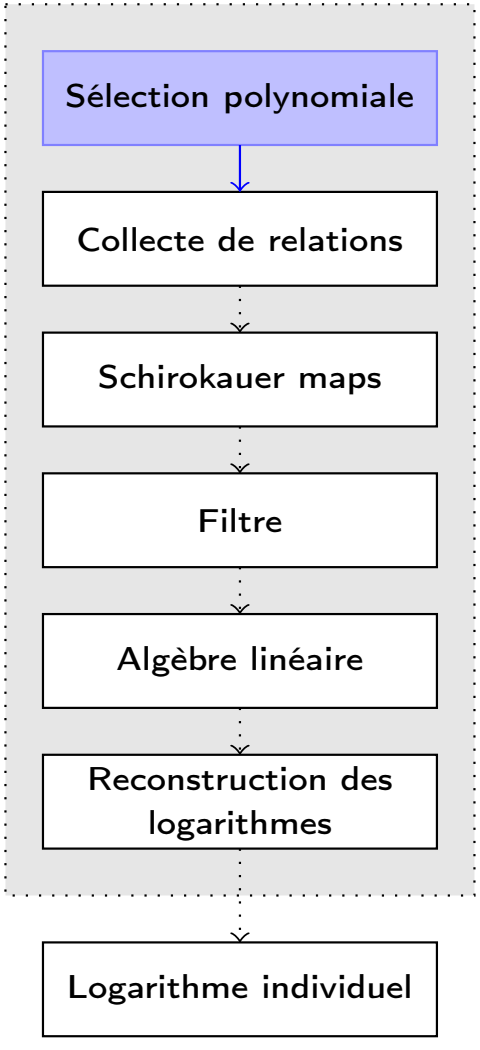
$$Norm_f((a,b)) = b^d f(\frac{a}{b})$$

$$Norm_g((a,b)) = b^d g(\frac{a}{b})$$

N est B-friable s'il se décompose en facteurs premiers plus petits que B :

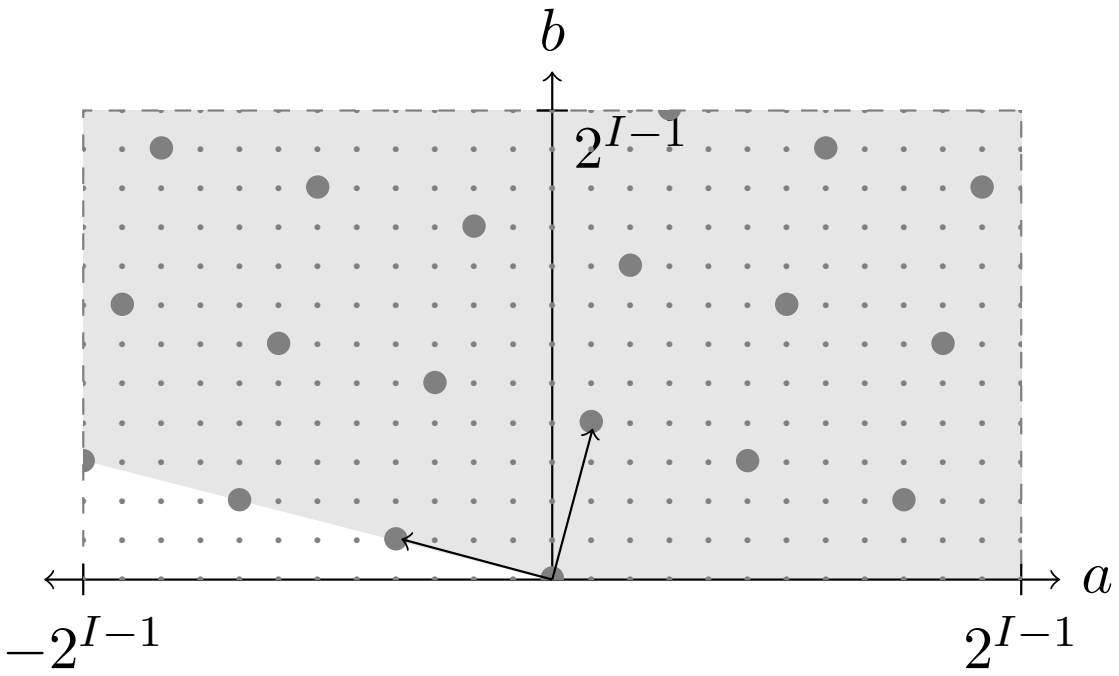
$$N = \prod_{q<B} q^{e_q}.$$

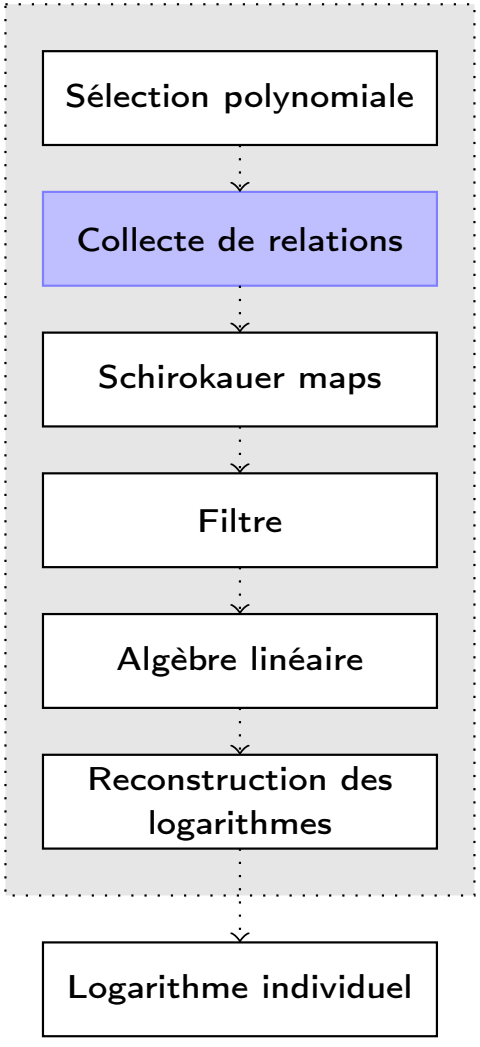
On veut exploiter la friabilité des normes.
Qualité des polynômes : quantité-E de Murphy.



CADO-NFS : Collecte de relations

On impose un spécial-$\mathfrak{q}$: facteur premier dans une relation.
On crible sur les normes des paires (a, b) dans un espace à deux dimensions :





CADO-NFS : Collecte de relations

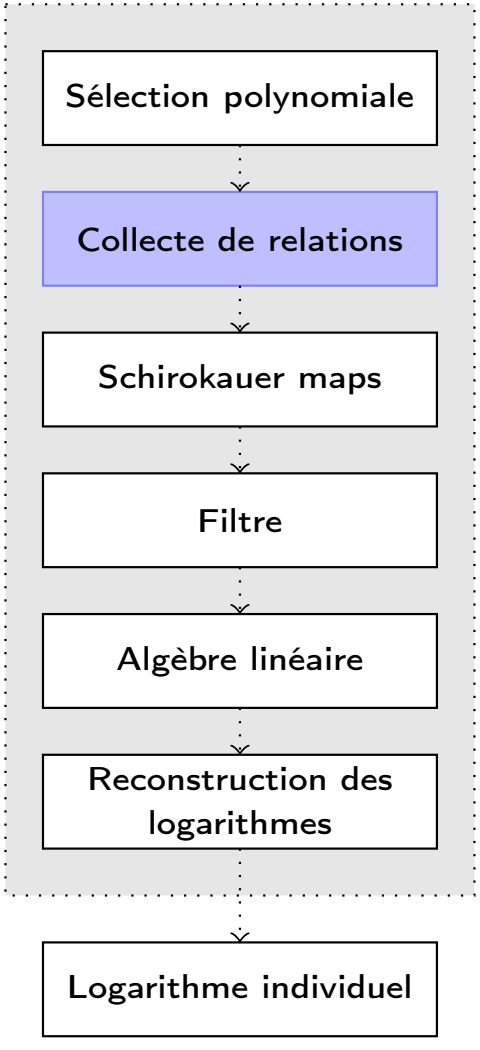
(a, b) = (201098, 29)
Exemple : $51094266926512316797053 9$

- Spécial-q
- Crible d'Eratosthène.

$$Norm((a,b)) = \left(\prod_{Norm(\mathfrak{q})<LIM} Norm(\mathfrak{q})^{e_{\mathfrak{q}}} \right) \times R.$$

- Cofactorisation par l'Elliptic Curve Method.

$$Norm((a,b)) = \prod_{Norm(\mathfrak{q})<LPB} Norm(\mathfrak{q})^{e_{\mathfrak{q}}}.$$



CADO-NFS : Collecte de relations

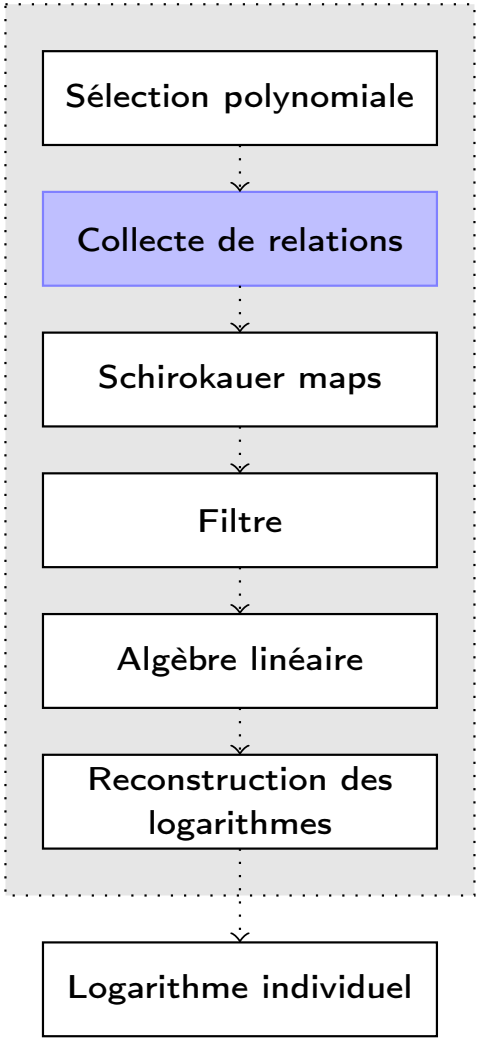
(a, b) = (201098, 29)
Exemple : $\mathbf{6217}.821847626291013620 67$

- Spécial-$\mathfrak{q}$
- Crible d'Eratosthène.

$$Norm((a,b)) = \left(\prod_{Norm(\mathfrak{q}) < LIM} Norm(\mathfrak{q})^{e_{\mathfrak{q}}} \right) \times R.$$

- Cofactorisation par l'Elliptic Curve Method.

$$Norm((a,b)) = \prod_{Norm(\mathfrak{q}) < LPB} Norm(\mathfrak{q})^{e_{\mathfrak{q}}}.$$



CADO-NFS : Collecte de relations

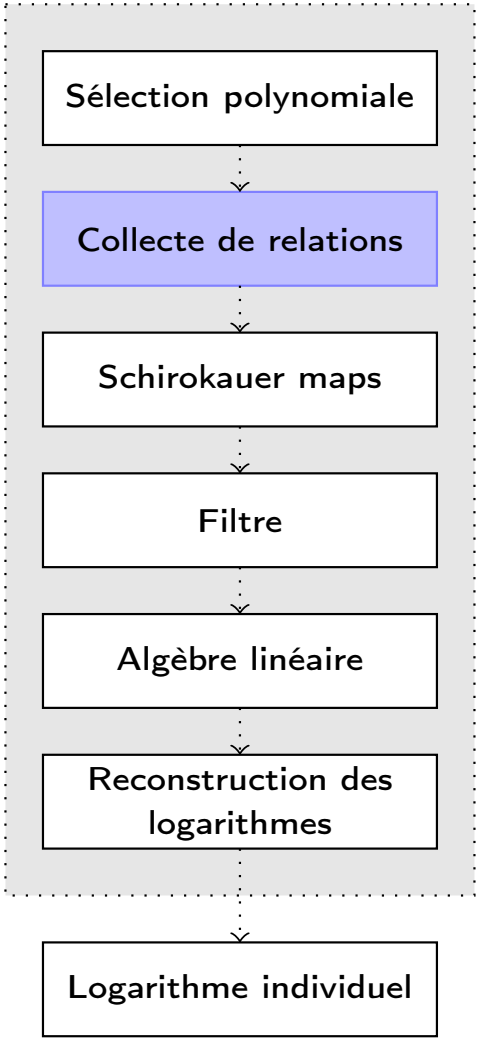
(a, b) = (201098, 29)
Exemple : $6217.\mathbf{3^3}.\mathbf{307}.\mathbf{1033}.\mathbf{11519}.833247589$

- Spécial-$\mathfrak{q}$
- Crible d'Eratosthène.

$$Norm\left((a,b)\right) = \left(\prod_{Norm(\mathfrak{q})<LIM} Norm\left(\mathfrak{q}\right)^{e_{\mathfrak{q}}}\right) \times R.$$

- Cofactorisation par l'Elliptic Curve Method.

$$Norm\left((a,b)\right) = \prod_{Norm(\mathfrak{q})<LPB} Norm\left(\mathfrak{q}\right)^{e_{\mathfrak{q}}}.$$



CADO-NFS : Collecte de relations

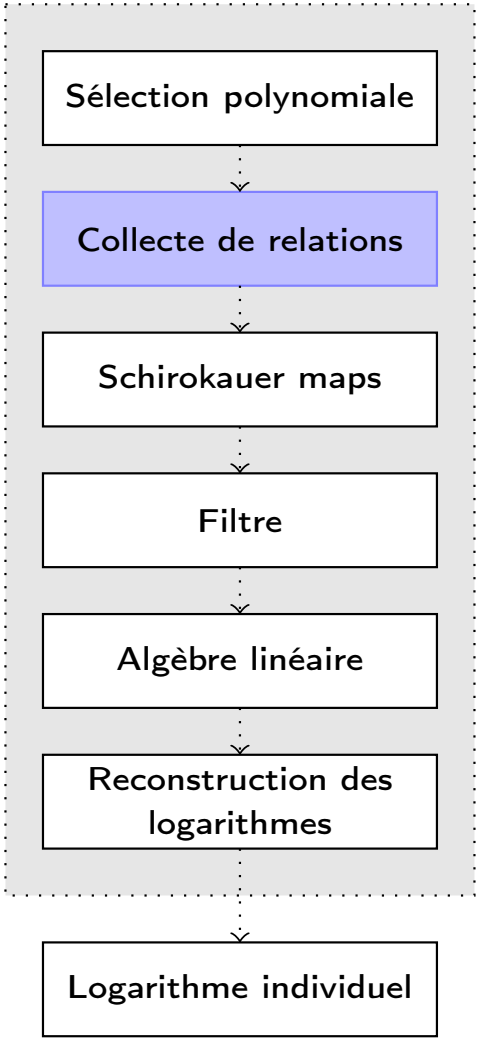
(a, b) = (201098, 29)
Exemple : $6217.3^3.307.1033.11519.\mathbf{27893.29873}$

- Spécial-$\mathfrak{q}$
- Crible d'Eratosthène.

$$Norm\left((a,b)\right) = \left(\prod_{Norm(\mathfrak{q})<LIM} Norm\left(\mathfrak{q}\right)^{e_{\mathfrak{q}}}\right) \times R.$$

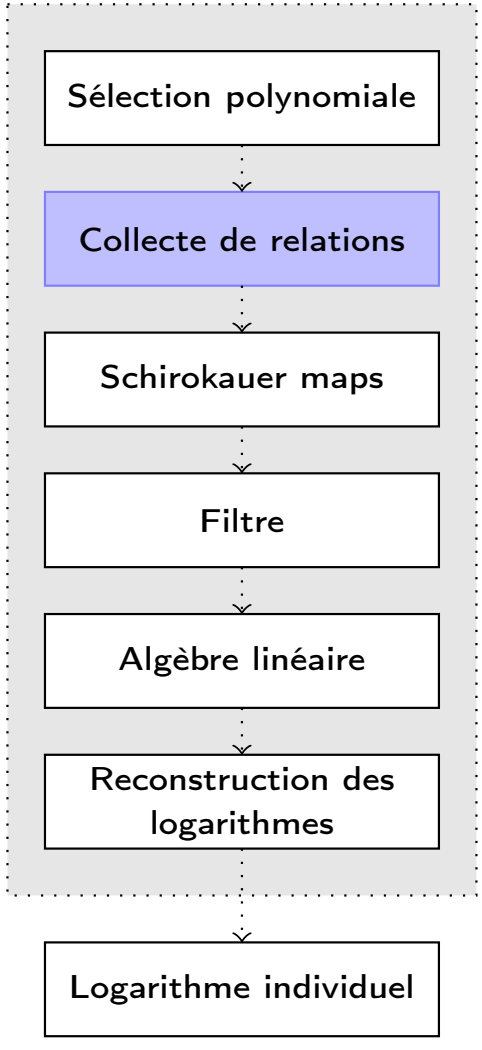
- Cofactorisation par l'Elliptic Curve Method.

$$Norm\left((a,b)\right) = \prod_{Norm(\mathfrak{q})<LPB} Norm\left(\mathfrak{q}\right)^{e_{\mathfrak{q}}}.$$



CADO-NFS : Collecte de relations

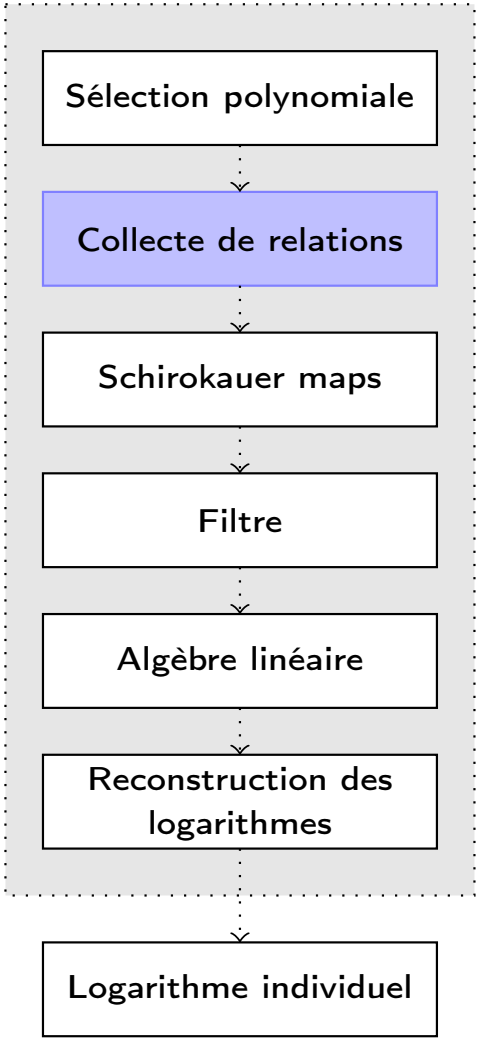
$$Norm\left((a,b)\right) = \prod_{Norm(\mathfrak{q})<LPB} Norm\left(\mathfrak{q}\right)^{e_{\mathfrak{q}}},$$



CADO-NFS : Collecte de relations

$$Norm\left((a,b)\right) = \prod_{Norm(\mathfrak{q})<LPB} Norm\left(\mathfrak{q}\right)^{e_{\mathfrak{q}}},$$

$$\text{“}\prod_i p_i^{e_i} \equiv \prod_j q_j^{d_j}\ [p]\text{”}^{1}.$$



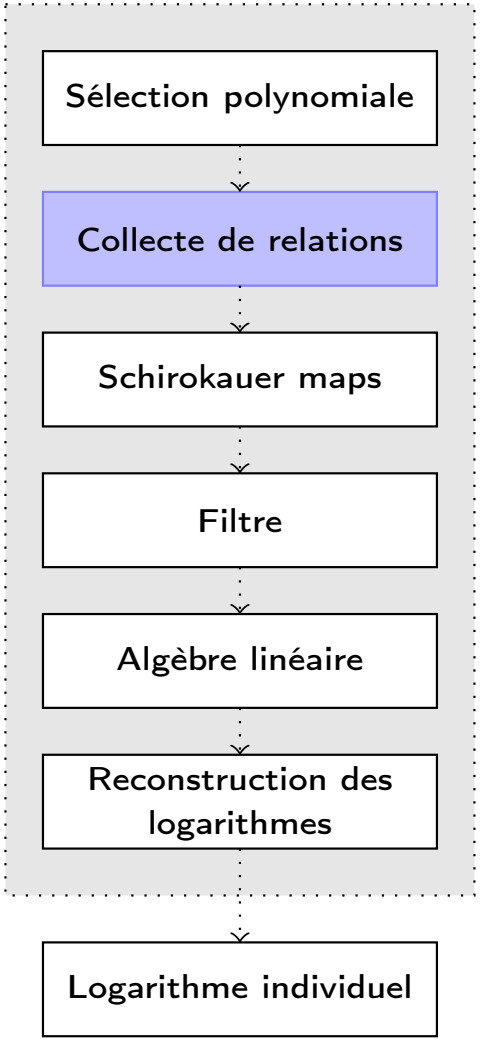
1 . C'est faux, pas de théorème de factorisation unique.

CADO-NFS : Collecte de relations

$$Norm\left((a,b)\right) = \prod_{Norm(\mathfrak{q})<LPB} Norm\left(\mathfrak{q}\right)^{e_{\mathfrak{q}}},$$

$$\text{“}\prod_i p_i^{e_i} \equiv \prod_j q_j^{d_j}\ [p]\text{”}^{1}.$$

$$\text{“}\sum_i e_i \log_g(p_i) - \sum_j d_j \log_g(q_j) \equiv 0\ [\ell]\text{”}^{1}.$$



1 . C'est faux, pas de théorème de factorisation unique.

CADO-NFS : Collecte de relations

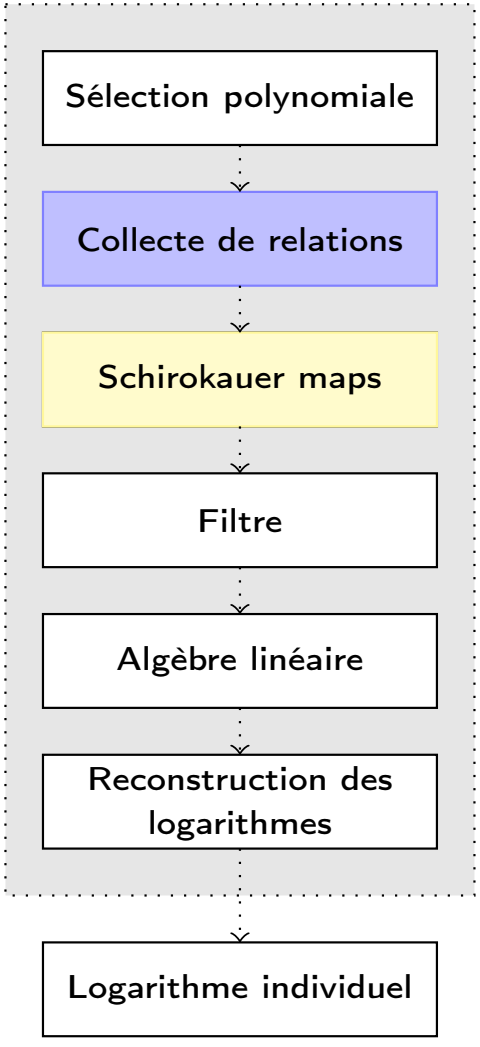
$$Norm\left((a,b)\right) = \prod_{Norm(\mathfrak{q})<LPB} Norm\left(\mathfrak{q}\right)^{e_{\mathfrak{q}}},$$

$$\text{“}\prod_i p_i^{e_i} \equiv \prod_j q_j^{d_j}\ [p]\text{”}^1.$$

$$\text{“}\sum_i e_i \log_g(p_i) - \sum_j d_j \log_g(q_j) \equiv 0\ [\ell]\text{”}^1.$$

$$\sum_i e_i \mathrm{vlog}_g(\mathfrak{p}_i) - \sum_j d_j \mathrm{vlog}_g(\mathfrak{q}_j) +$$

$$\sum_k SM_k\left(a,b\right) \mathrm{vlog}_g(SM_k) \equiv 0\ [\ell]\,.$$

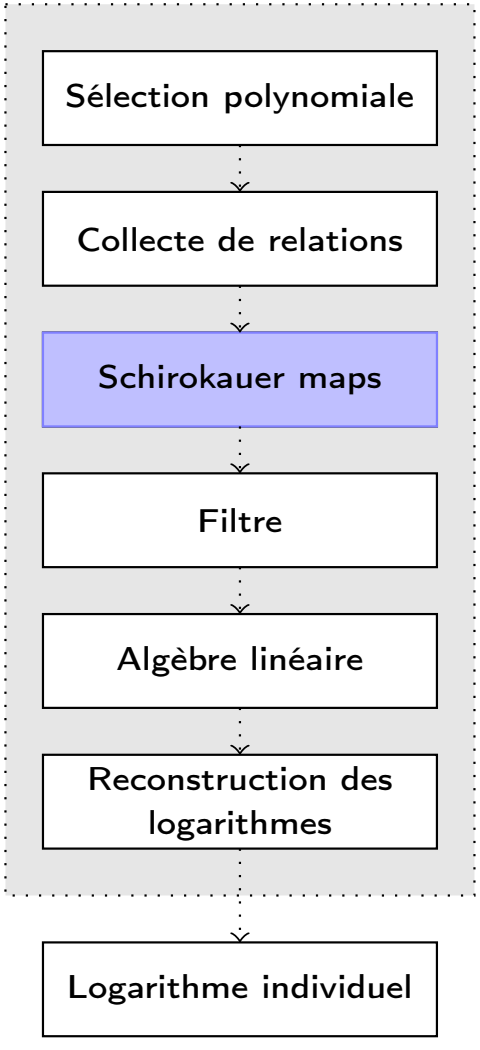


CADO-NFS : Algèbre linéaire

$$\sum_i e_i \mathrm{vlog}_g(\mathfrak{p}_i) - \sum_j d_j \mathrm{vlog}_g(\mathfrak{q}_j) + \sum_k SM_k\,(a,b)\,\mathrm{vlog}_g(SM_k) \equiv 0\,[\ell]$$

Système d'équations linéaires.
Les logarithmes sont les inconnues.
Les exposants sont les coefficients de la matrice.

$$\begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n-1} & a_{1,n} & SM_{1,1} & \cdots & SM_{1,m} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n-1} & a_{2,n} & SM_{2,1} & \cdots & SM_{2,m} \\ \vdots & & & & & & & \vdots \\ \vdots & & & & & & & \vdots \\ \vdots & & & & & & & \vdots \\ a_{k,1} & a_{k,2} & \cdots & a_{k,n-1} & a_{k,n} & SM_{k,1} & \cdots & SM_{k,m} \end{pmatrix}$$

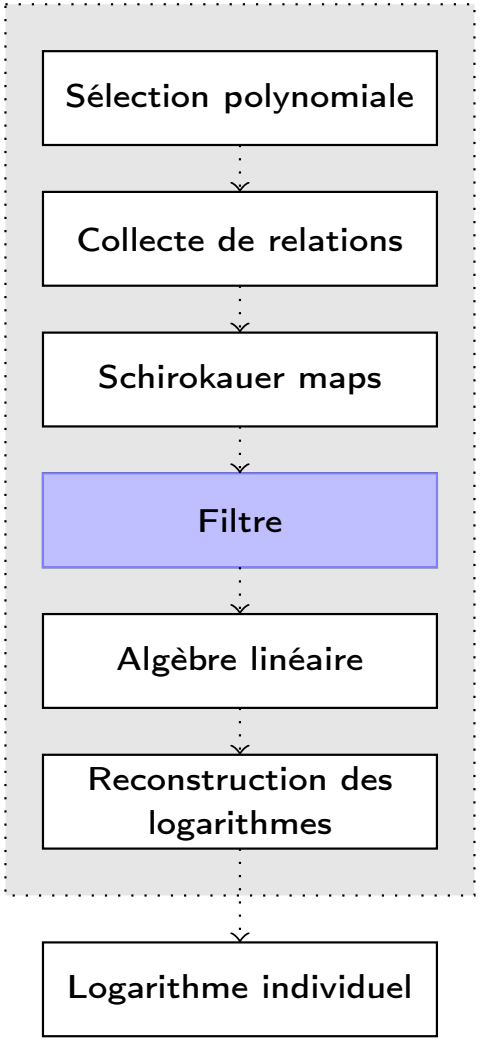


CADO-NFS : Algèbre linéaire

$$\sum_i e_i \mathrm{vlog}_g(\mathfrak{p}_i) - \sum_j d_j \mathrm{vlog}_g(\mathfrak{q}_j) + \sum_k SM_k\,(a,b)\,\mathrm{vlog}_g(SM_k) \equiv 0\,[\ell]$$

Système d'équations linéaires.
Les logarithmes sont les inconnues.
Les exposants sont les coefficients de la matrice.
Élimination Gaussienne.

$$\begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n-1} & a_{1,n} & SM_{1,1} & \cdots & SM_{1,m} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n-1} & a_{2,n} & SM_{2,1} & \cdots & SM_{2,m} \\ \vdots & & & & & & & \vdots \\ \vdots & & & & & & & \vdots \\ \vdots & & & & & & & \vdots \\ a_{k,1} & a_{k,2} & \cdots & a_{k,n-1} & a_{k,n} & SM_{k,1} & \cdots & SM_{k,m} \end{pmatrix}$$

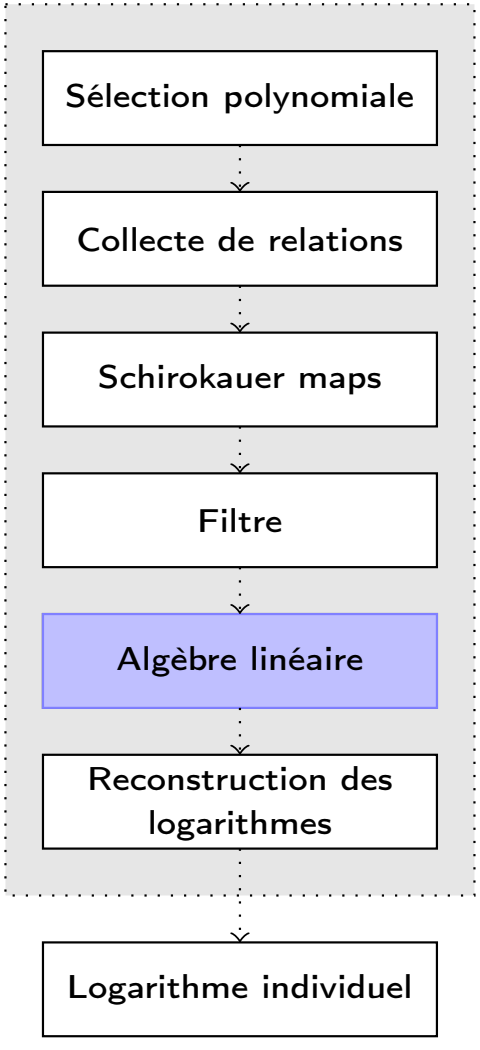


CADO-NFS : Algèbre linéaire

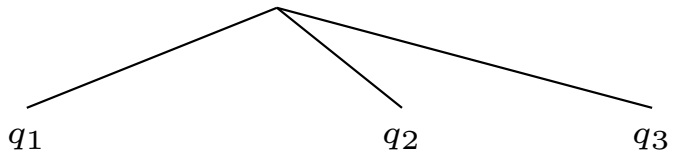
$$\sum_i e_i \mathrm{vlog}_g(\mathfrak{p}_i) - \sum_j d_j \mathrm{vlog}_g(\mathfrak{q}_j) + \sum_k SM_k\,(a,b)\,\mathrm{vlog}_g(SM_k) \equiv 0\,[\ell]$$

Système d'équations linéaires.
Les logarithmes sont les inconnues.
Les exposants sont les coefficients de la matrice.
Élimination Gaussienne.
Block-Wiedemann.

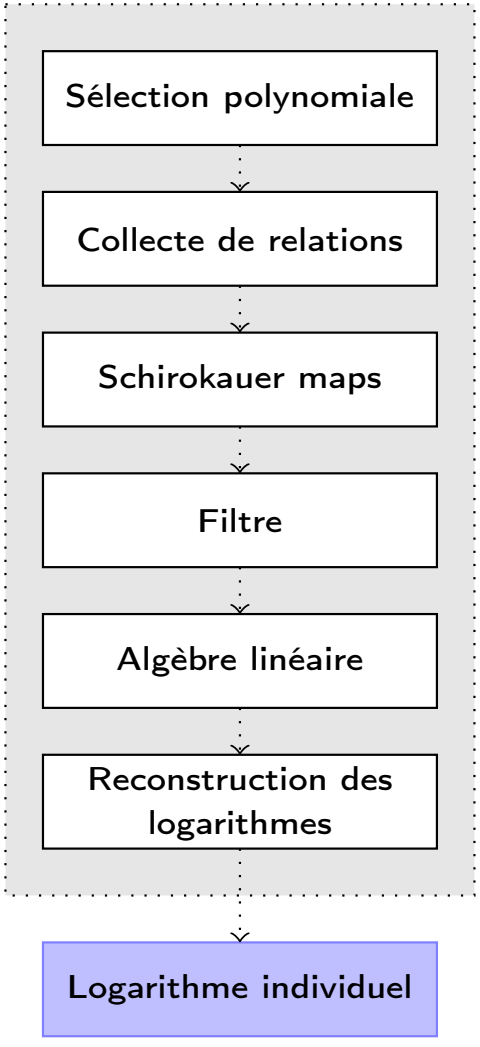
$$\begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n-1} & a_{1,n} & SM_{1,1} & \cdots & SM_{1,m} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n-1} & a_{2,n} & SM_{2,1} & \cdots & SM_{2,m} \\ \vdots & & & & & & & \vdots \\ \vdots & & & & & & & \vdots \\ \vdots & & & & & & & \vdots \\ a_{k,1} & a_{k,2} & \cdots & a_{k,n-1} & a_{k,n} & SM_{k,1} & \cdots & SM_{k,m} \end{pmatrix}$$



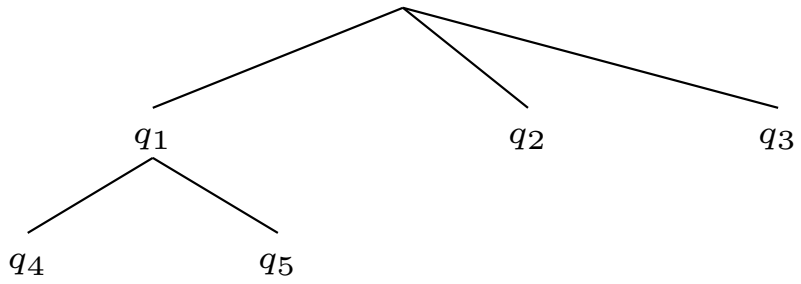
CADO-NFS : Logarithmes individuels



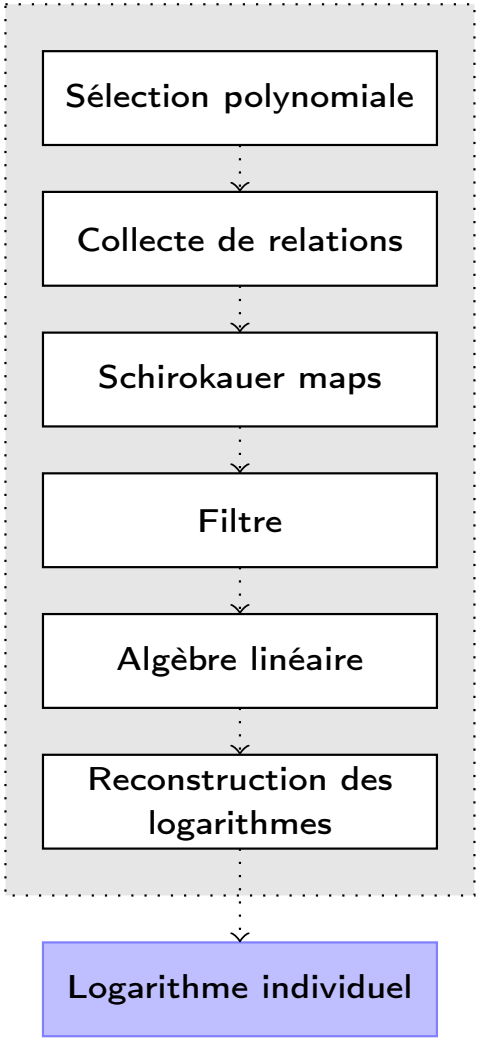
Base de donnée des logarithmes



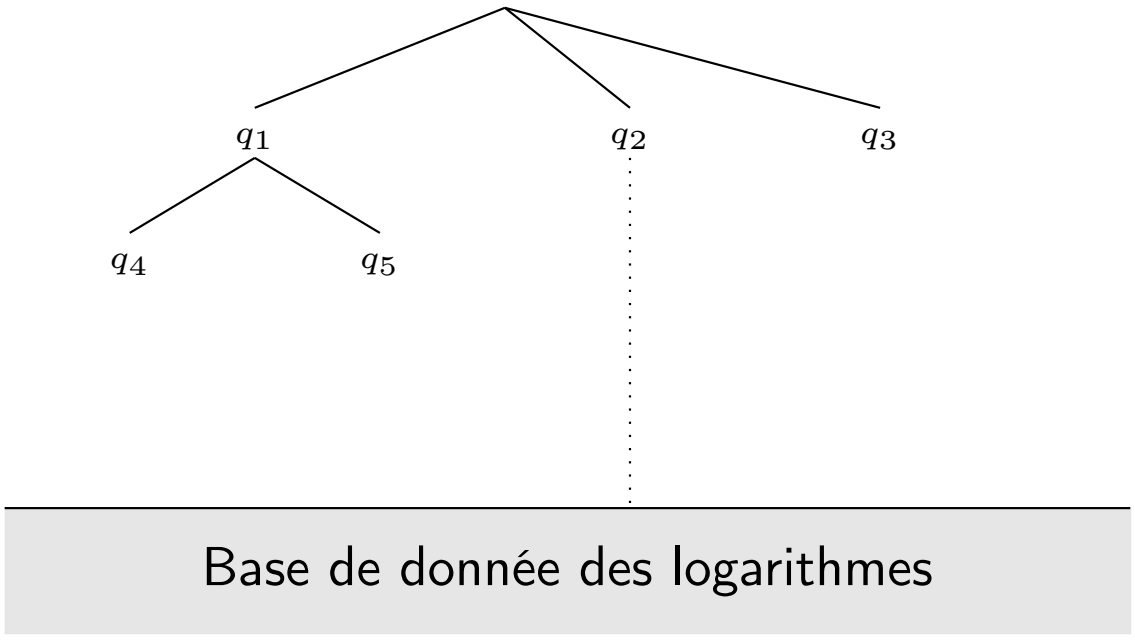
CADO-NFS : Logarithmes individuels

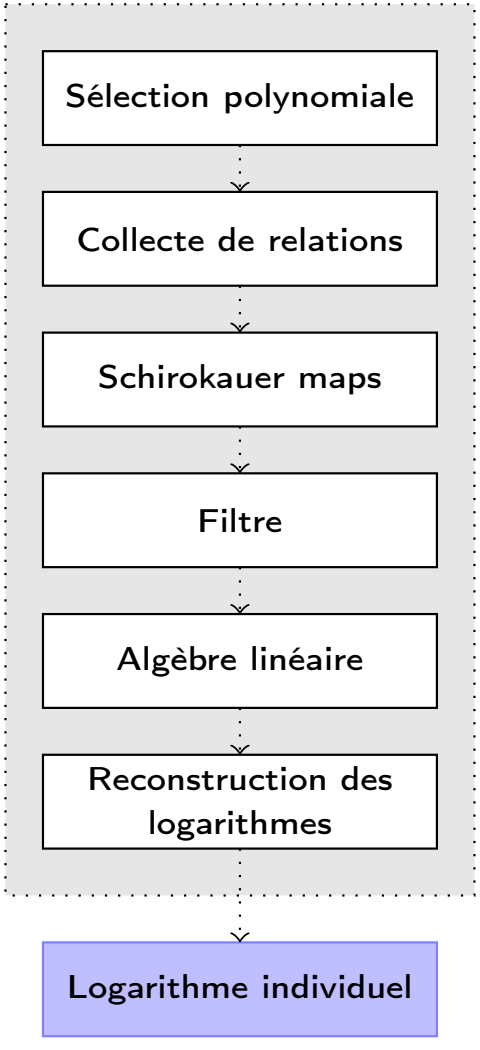


Base de donnée des logarithmes

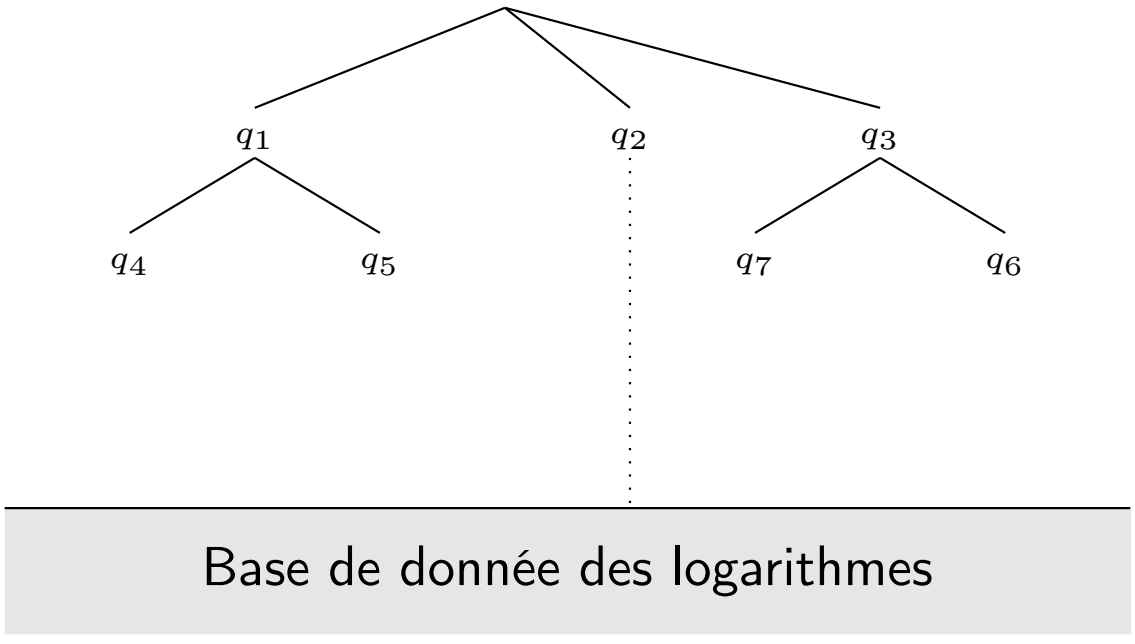


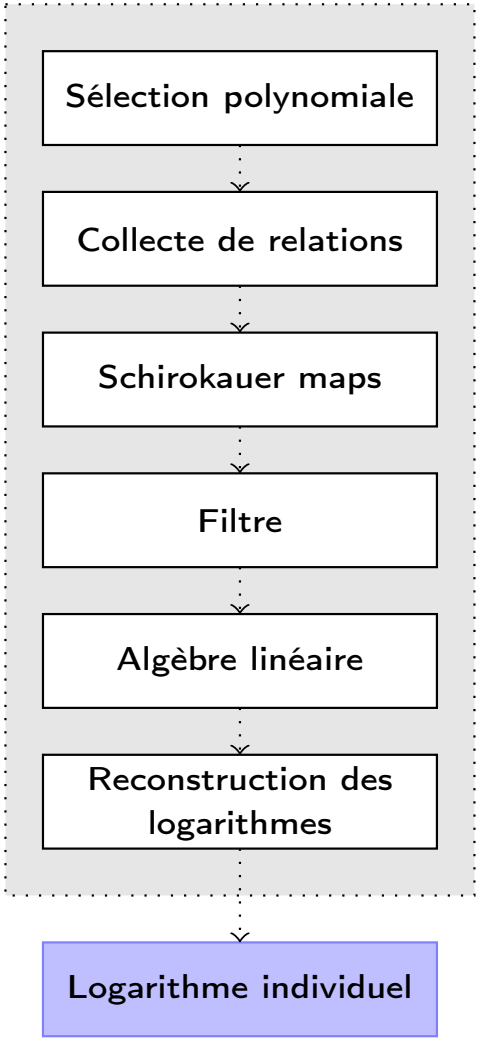
CADO-NFS : Logarithmes individuels



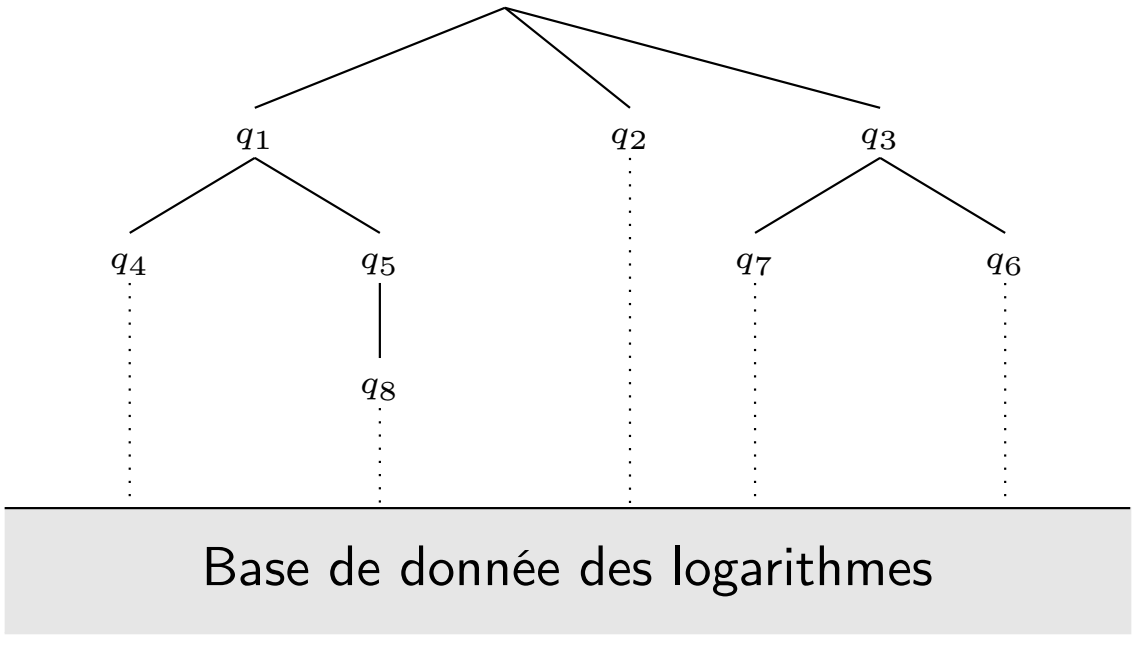


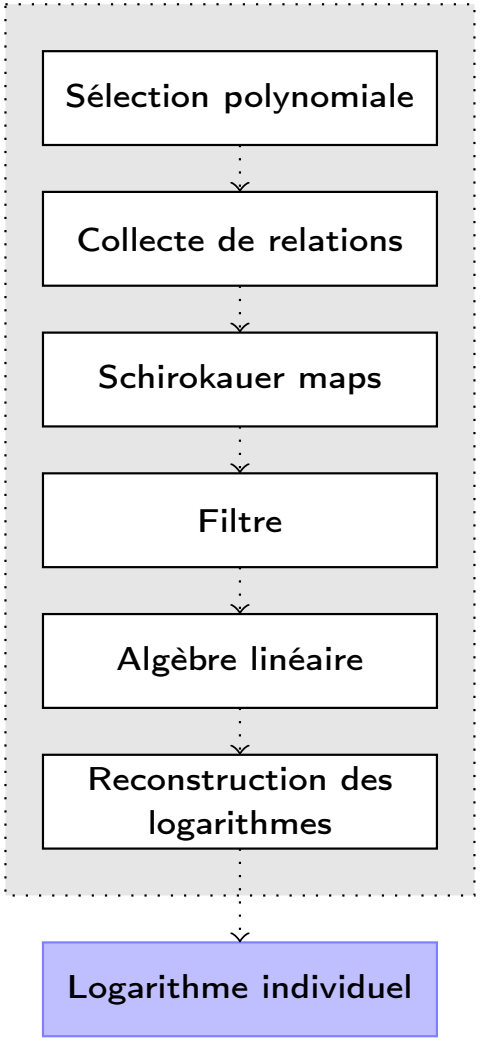
CADO-NFS : Logarithmes individuels





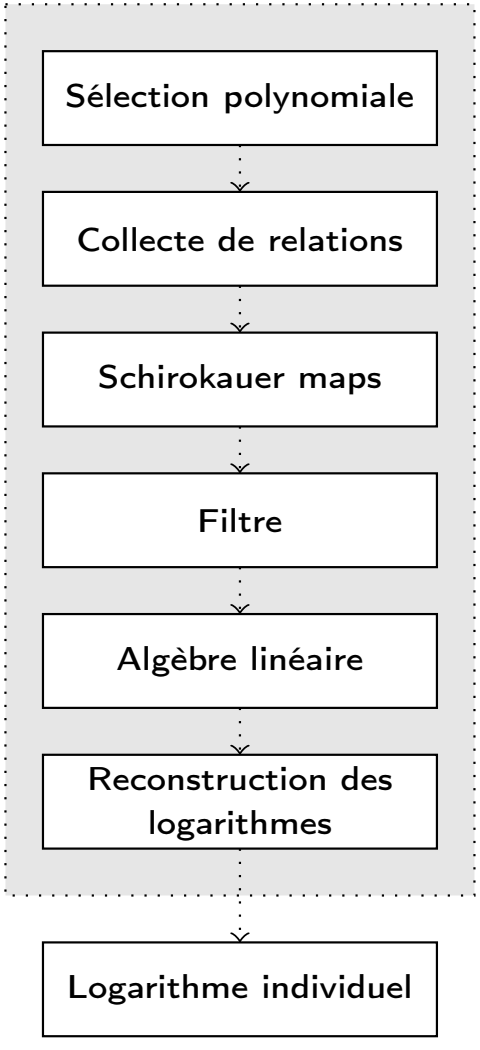
CADO-NFS : Logarithmes individuels



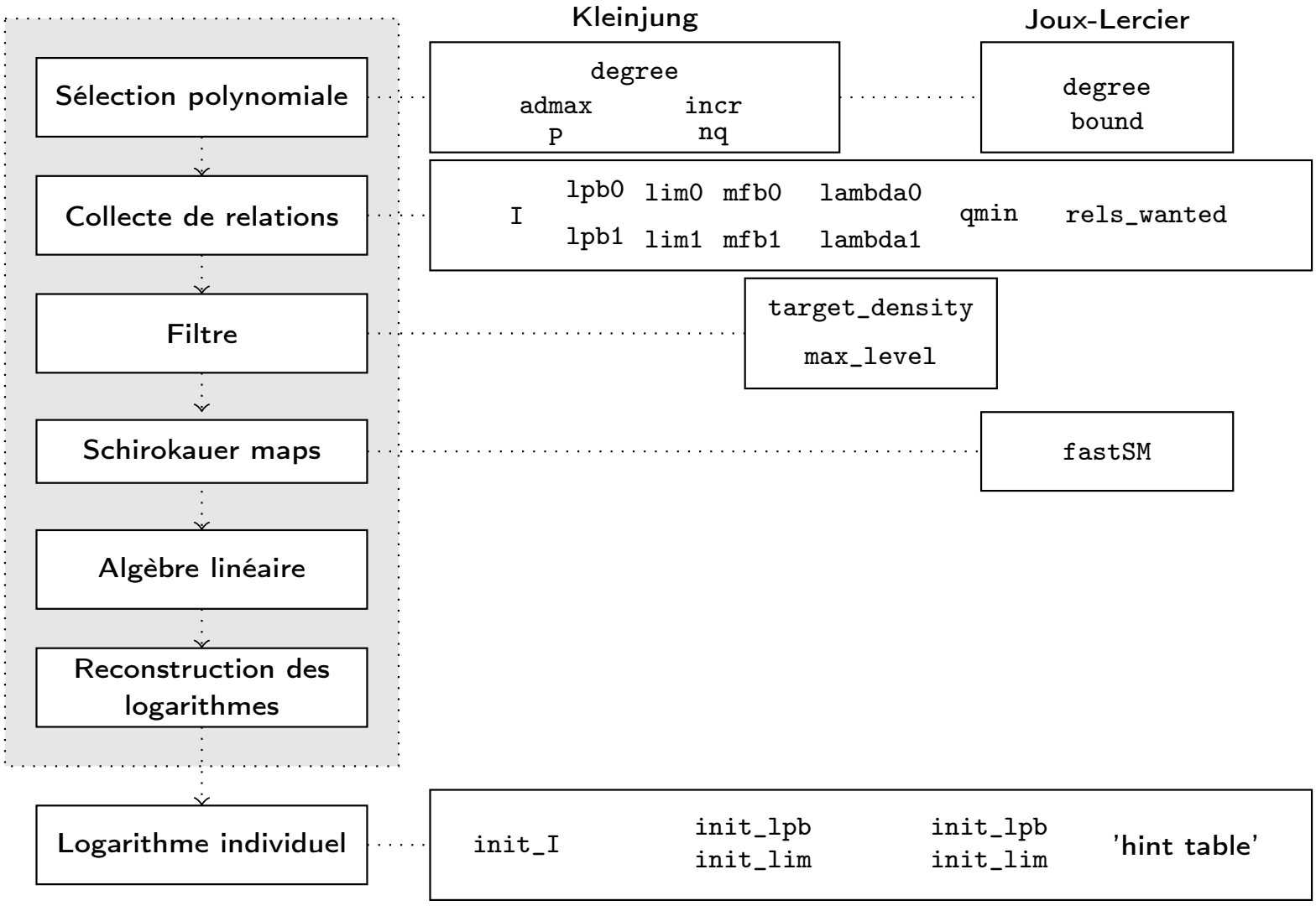


CADO-NFS en bref

- On sélectionne des polynômes.
- On factorise les normes pour obtenir des relations.
- On fabrique un système linéaire où les logarithmes sont les inconnues.
- On résout le système par l'algèbre linéaire.
- On fabrique un arbre pour exprimer un logarithme cible en fonction de logarithmes connus et précalculés.



Vue d'ensemble des principaux paramètres



CADO-NFS : Idées pour le paramétrage

Le pire cas : p nombre premier de Sophie Germain.
C'est-à-dire $p = 2\ell + 1$ avec ℓ premier.
Étude de NFS dans le cas p de Sophie Germain et $\ell = \frac{p-1}{2}$.

- Première approche : s'intéresser au temps CPU total.
  - Trop de temps d'attente.
  - Peu d'informations sur la pertinence des paramètres obtenus.
- Seconde approche : s'intéresser aux résultats des différentes étapes.
  - Permet interprétation locale des résultats.
  - Mais pas de garantie sur l'optimalité.

CADO-NFS : Idées pour le paramétrage

Le pire cas : p nombre premier de Sophie Germain.
C'est-à-dire $p = 2\ell + 1$ avec ℓ premier.
Étude de NFS dans le cas p de Sophie Germain et $\ell = \frac{p-1}{2}$.

- Première approche : s'intéresser au temps CPU total.
  - Trop de temps d'attente.
  - Peu d'informations sur la pertinence des paramètres obtenus.
- Seconde approche : s'intéresser aux résultats des différentes étapes.
  - Permet interprétation locale des résultats.
  - Mais pas de garantie sur l'optimalité.

CADO-NFS : Idées pour le paramétrage

Le pire cas : p nombre premier de Sophie Germain.
C'est-à-dire $p = 2\ell + 1$ avec ℓ premier.
Étude de NFS dans le cas p de Sophie Germain et $\ell = \frac{p-1}{2}$.

- Première approche : s'intéresser au temps CPU total.
  - Trop de temps d'attente.
  - Peu d'informations sur la pertinence des paramètres obtenus.
- Seconde approche : s'intéresser aux résultats des différentes étapes.
  - Permet interprétation locale des résultats.
  - Mais pas de garantie sur l'optimalité.

Exemple : paramétrage de la collecte de relations

Étude avec `las` : unités de travail indépendantes.
Paramètres :

```
lpb0      lpb1
lim0      lim1
mfb0      mfb1
lambda0   lambda1
```

```
Total 9054 reports [0.00128s/r, 18.9r/sq] in 13.6
elapsed s [85.2% CPU]
```

Score :

$$\frac{\text{relations par spécial-q}}{\text{temps écoulé} \times \text{pourcentage de temps CPU}}$$

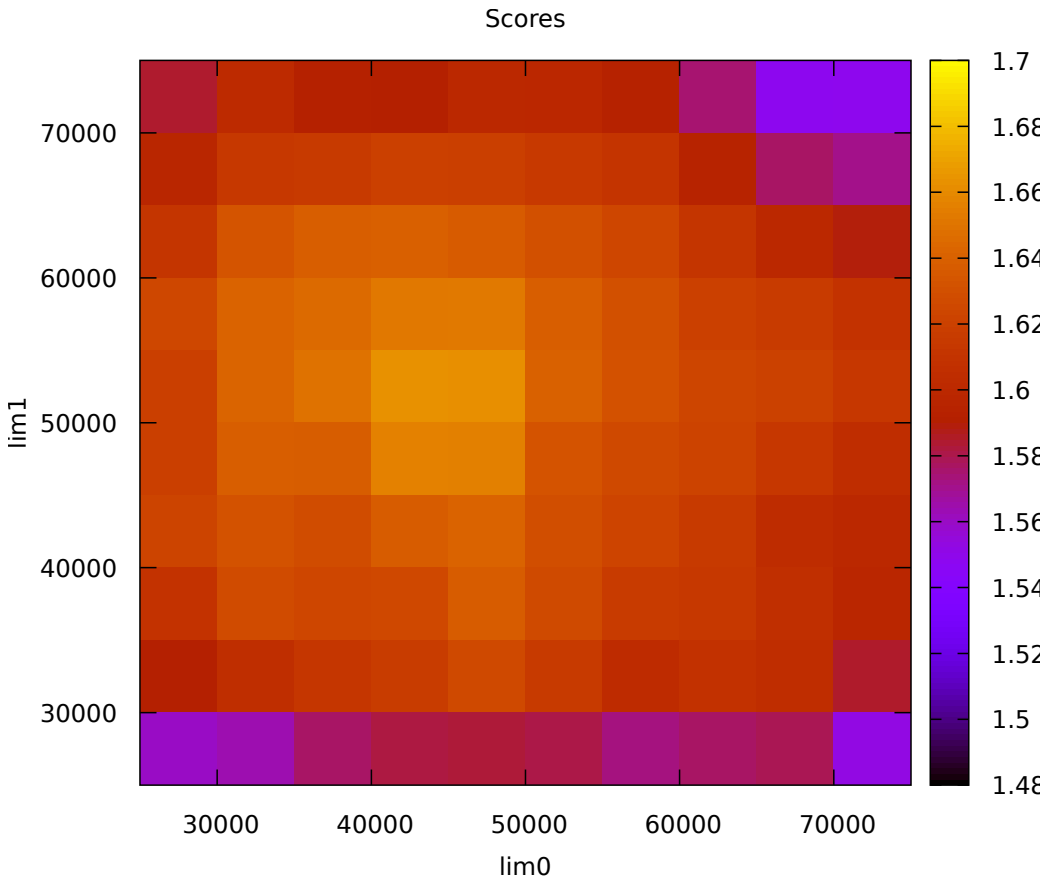
Exemple : paramétrage de la collecte de relations

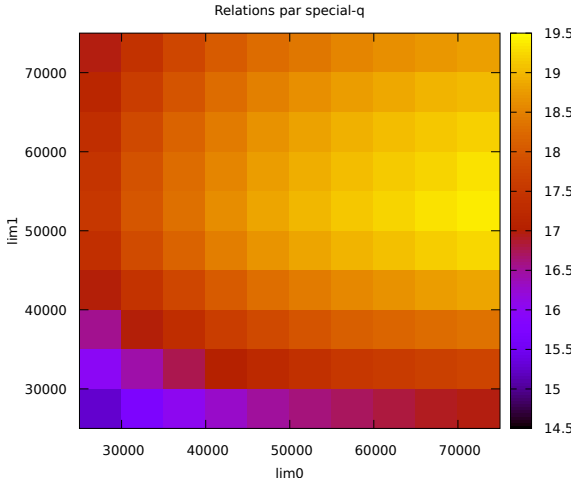
Score général :

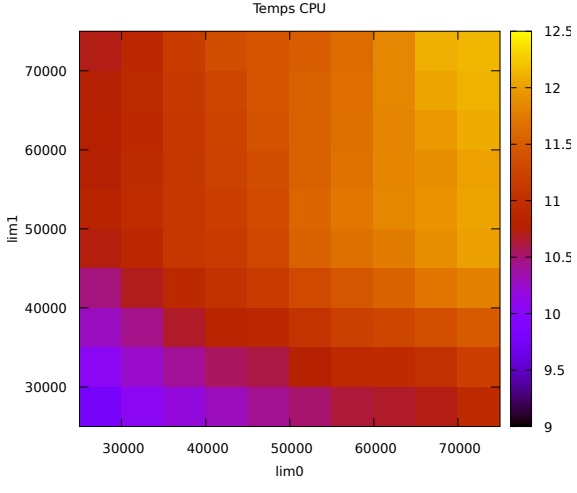
$$\frac{\text{score}}{\pi\left(2^{\texttt{lpb0}}\right)+\pi\left(2^{\texttt{lpb1}}\right)}.$$

Rang	`lpb`	`lim`	rels/spécial-q	score	score général
1	18, 18	52428, 52428	19	1.66138	3.61169×10^{-5}
2	19, 19	52428, 52428	39.8	2.95939	3.41023×10^{-5}
3	18, 19	52428, 52428	27.9	2.24091	3.37537×10^{-5}
⋮	⋮	⋮	⋮	⋮	⋮
9	17, 17	26214, 39321	6	0.65337	2.66659×10^{-5}
⋮	⋮	⋮	⋮	⋮	⋮
15	20, 20	314572, 104857	72.5	3.46641	2.11302×10^{-5}
⋮	⋮	⋮	⋮	⋮	⋮
23	16, 16	26214, 32768	1.6	0.19216	1.46863×10^{-5}

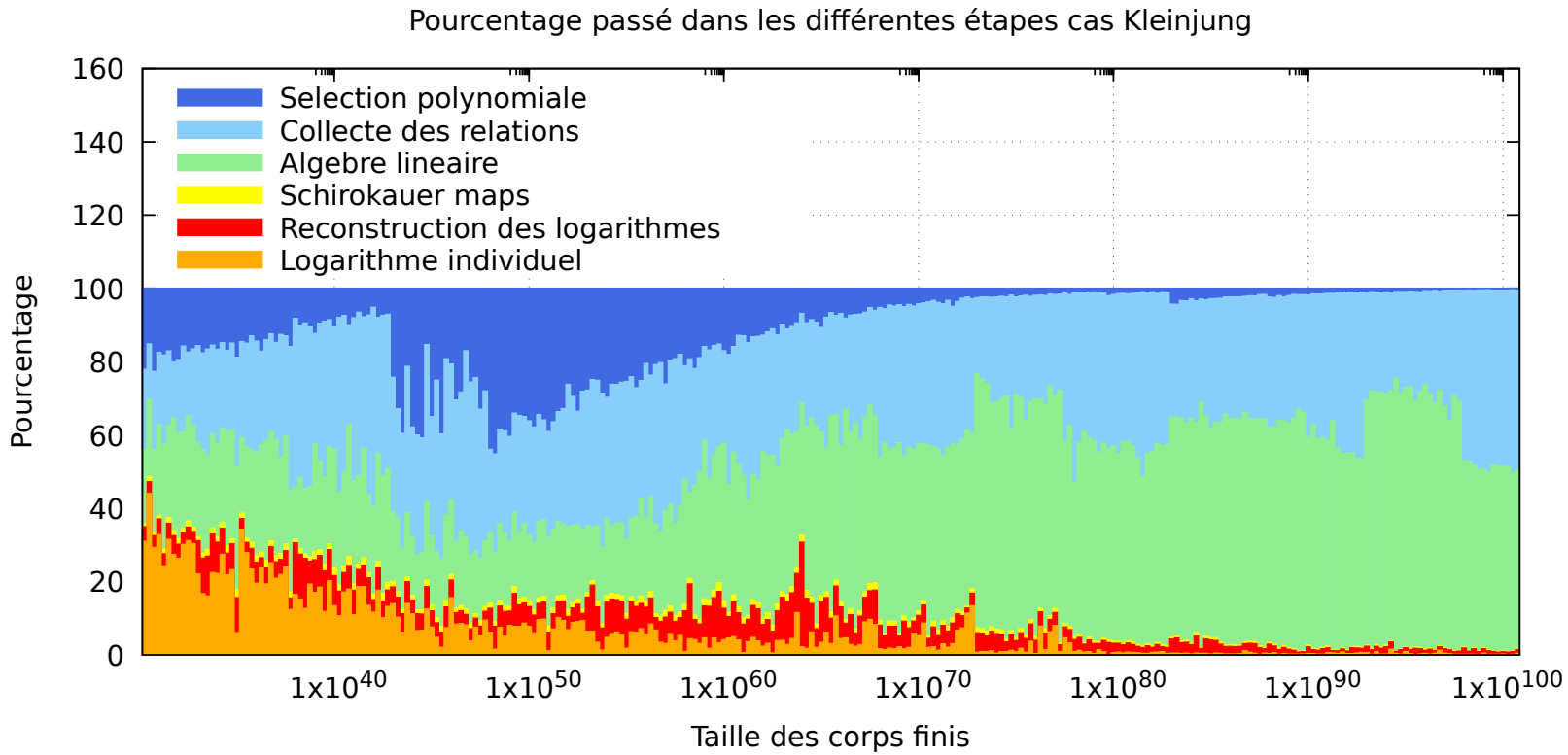
Exemple : paramétrage de la collecte de relations



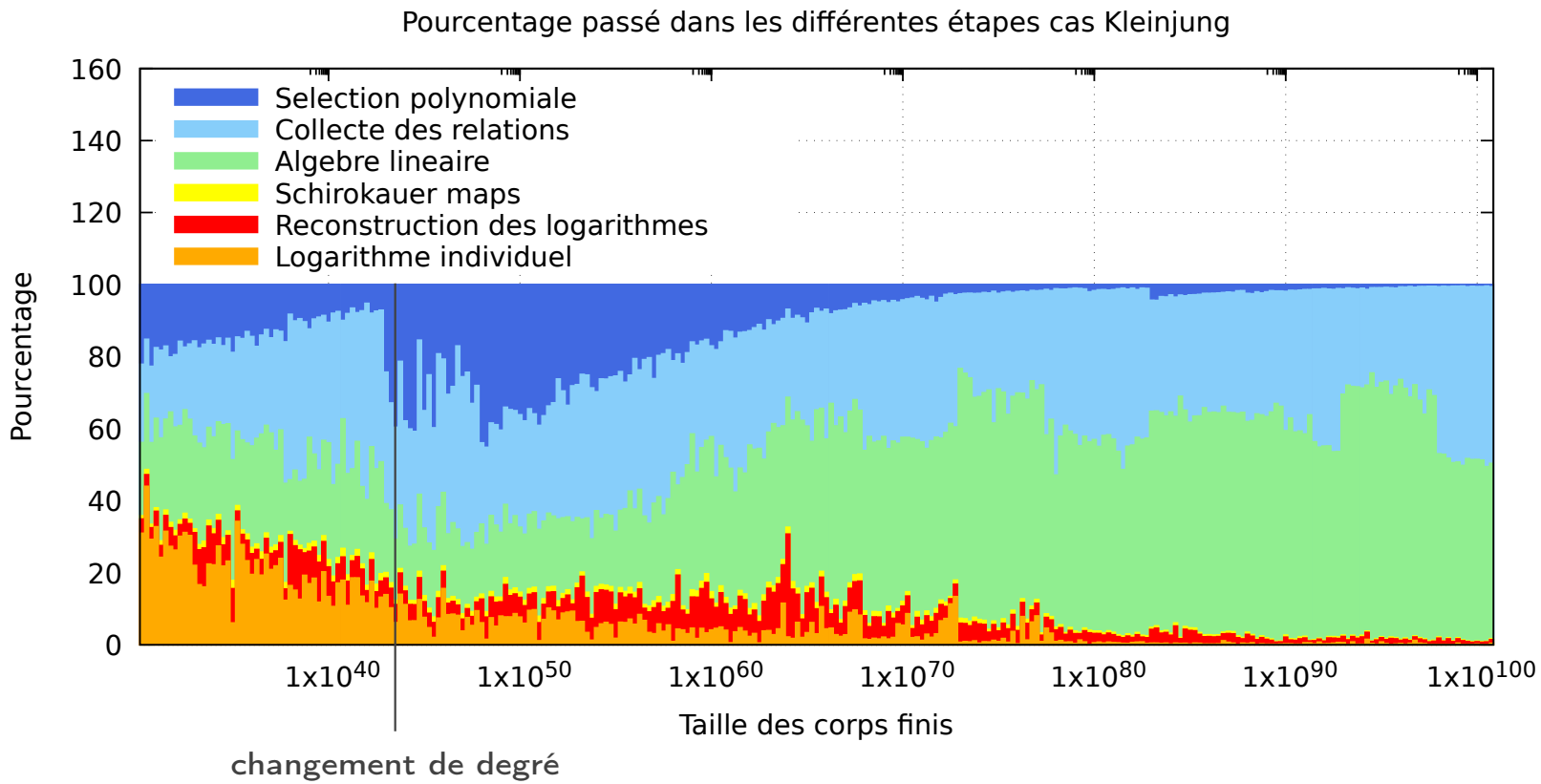




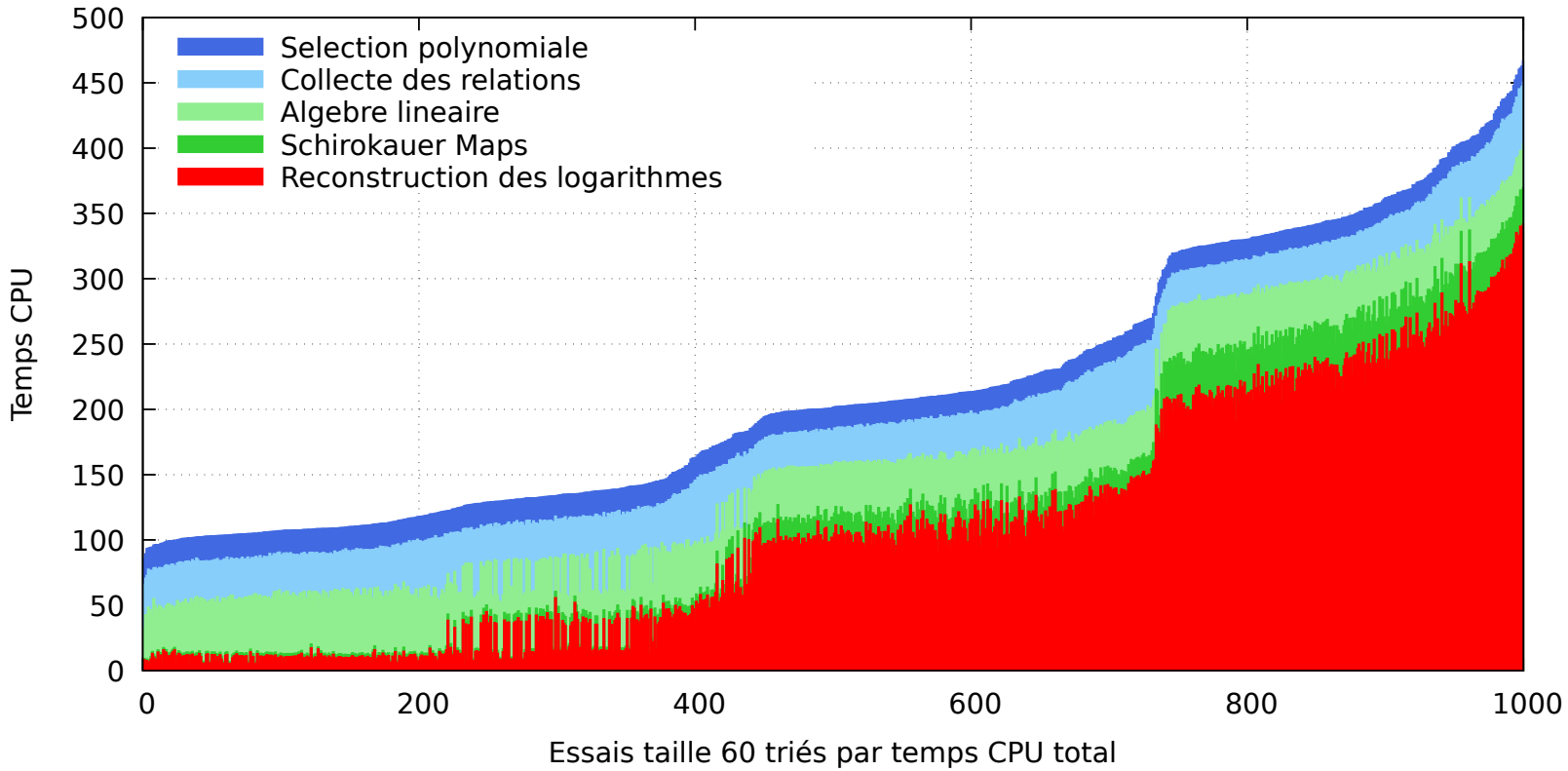
Exemple : Temps et petites tailles



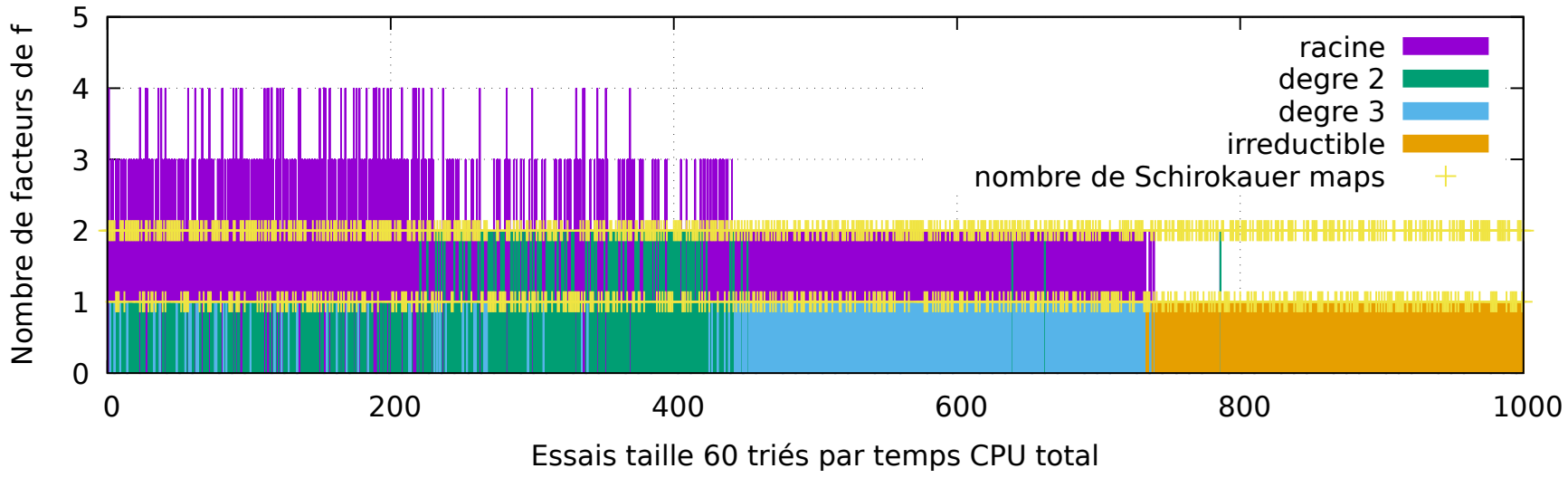
Exemple : Temps et petites tailles



Problème des Schirokauer maps



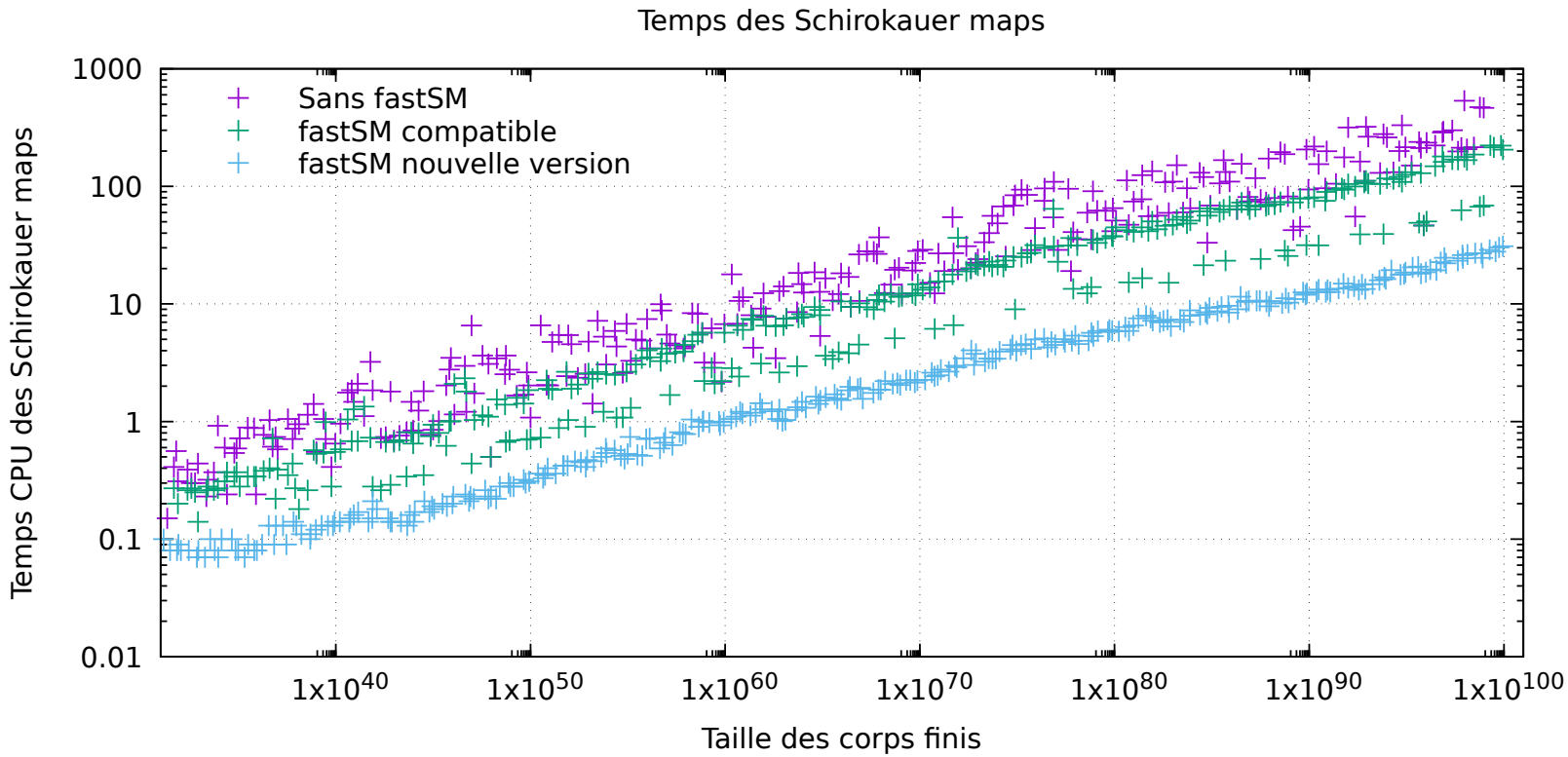
Degré des facteurs



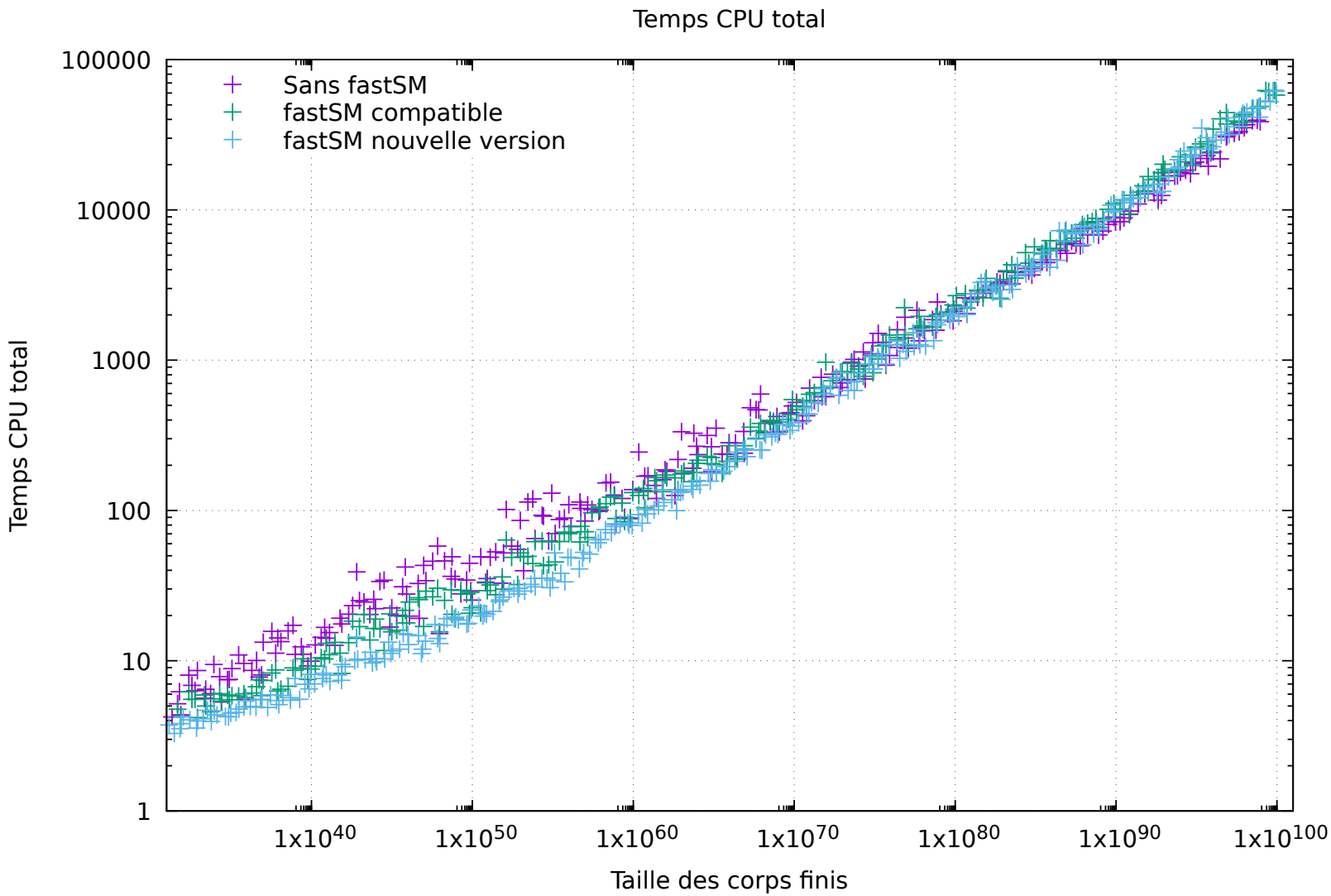
Exemple si $f = f_1.f_2.f_3$ dans $\mathbb{F}_\ell[X]$, alors

$$\mathbb{F}_\ell[X]/f(X) \simeq \mathbb{F}_\ell[X]/f_1(X) \times \mathbb{F}_\ell[X]/f_2(X) \times \mathbb{F}_\ell[X]/f_3(X).$$

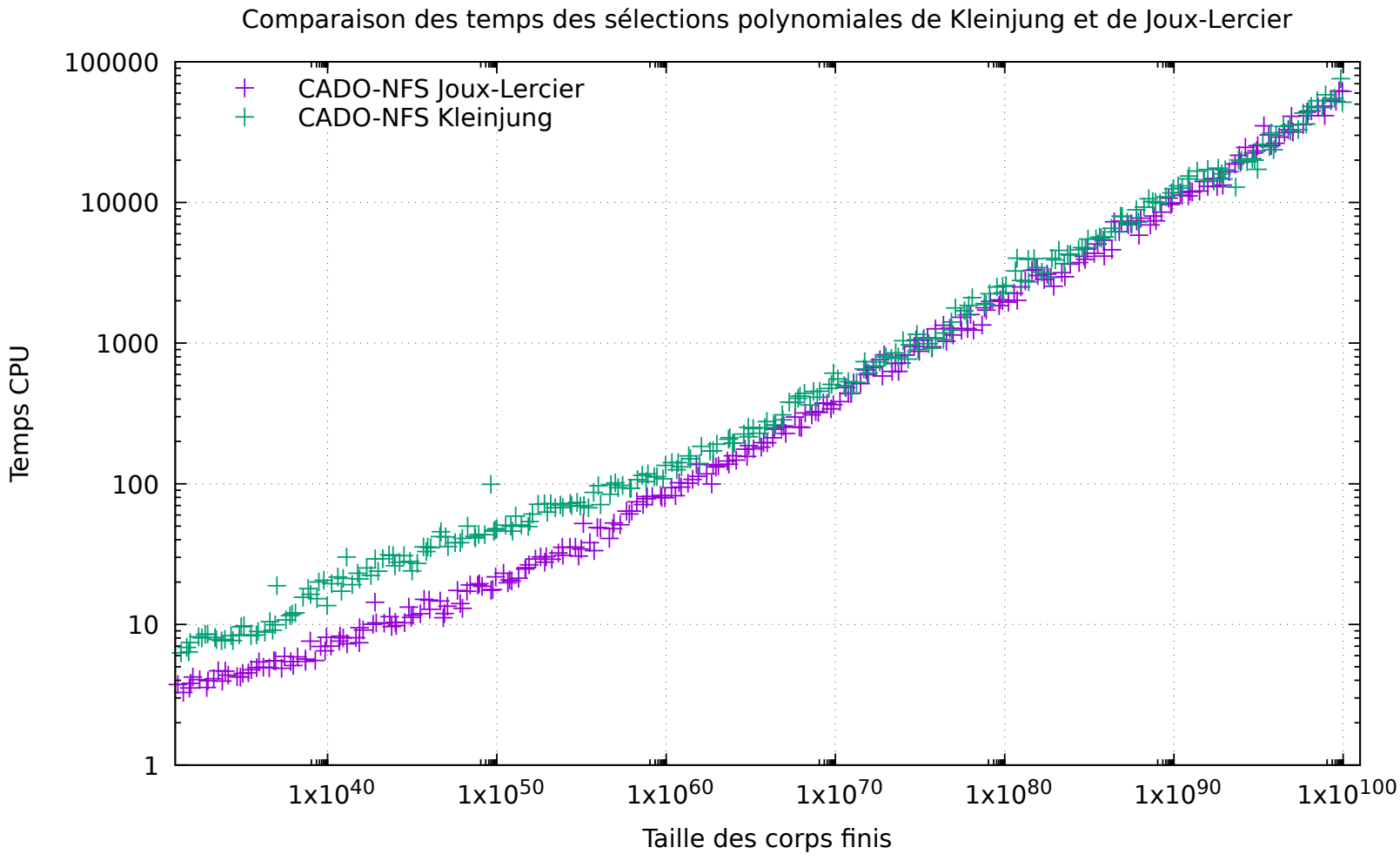
Solutions au problème des Schirokauer maps



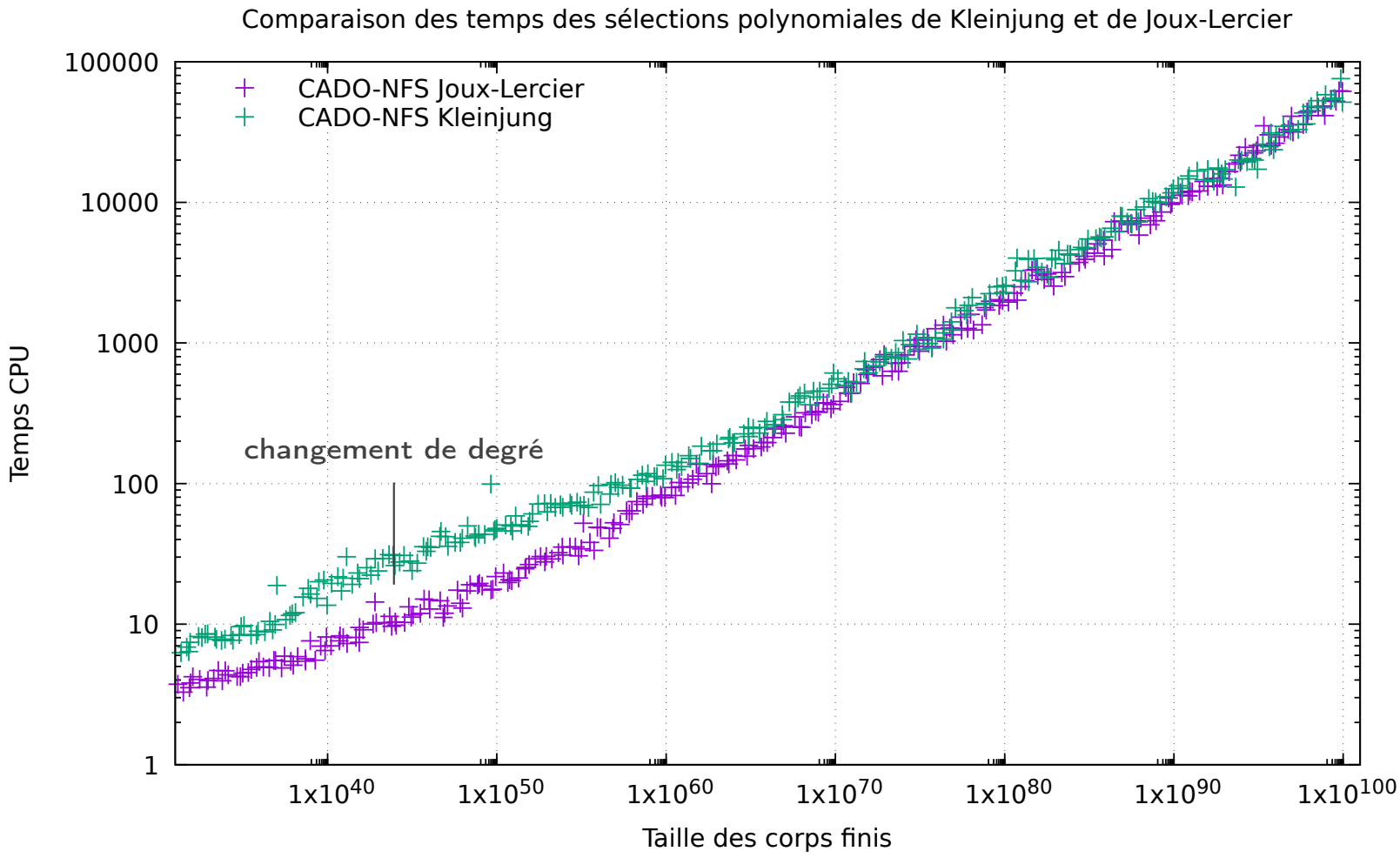
Solutions au problème des Schirokauer maps



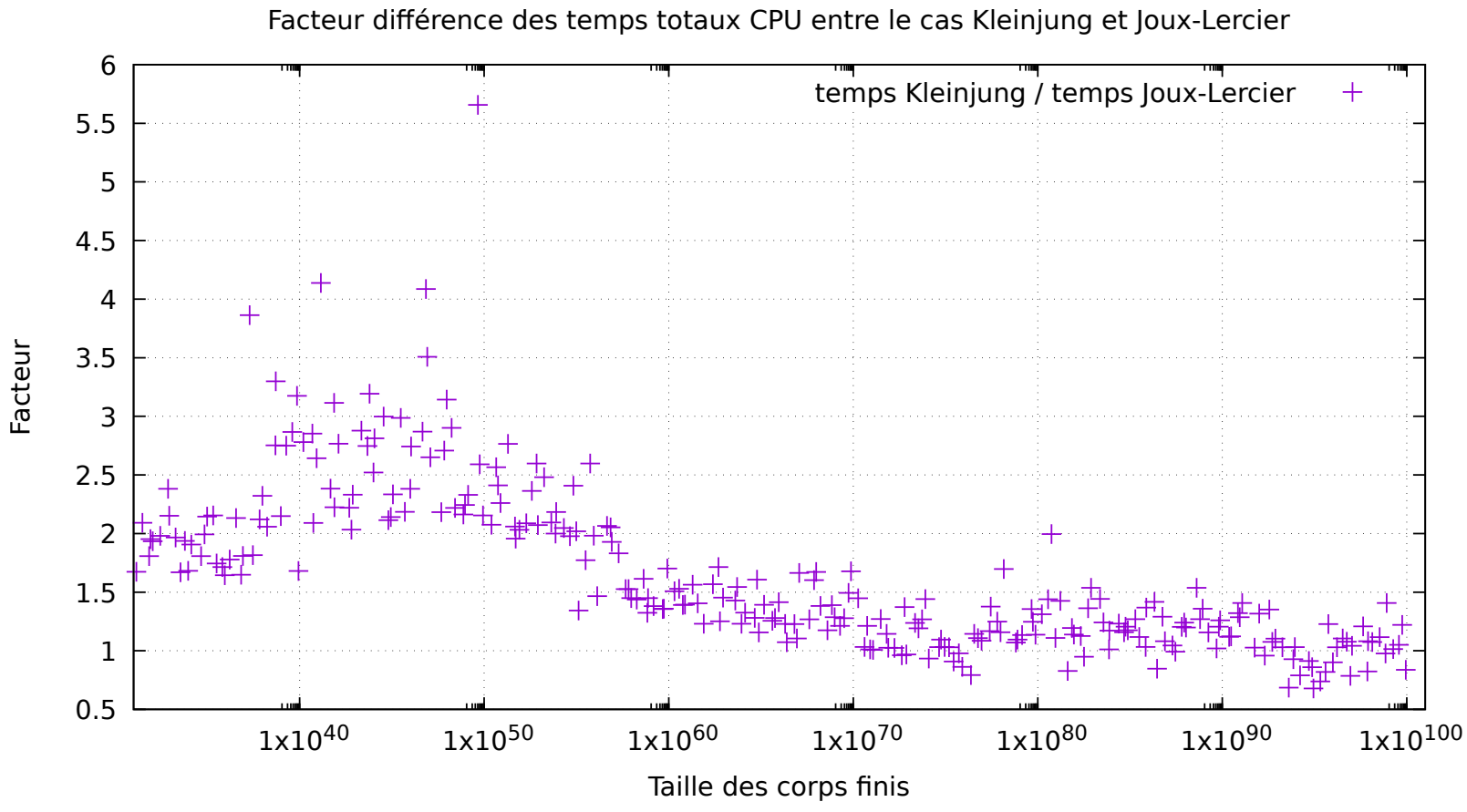
Sélection polynomiale Joux-Lercier et Kleinjung



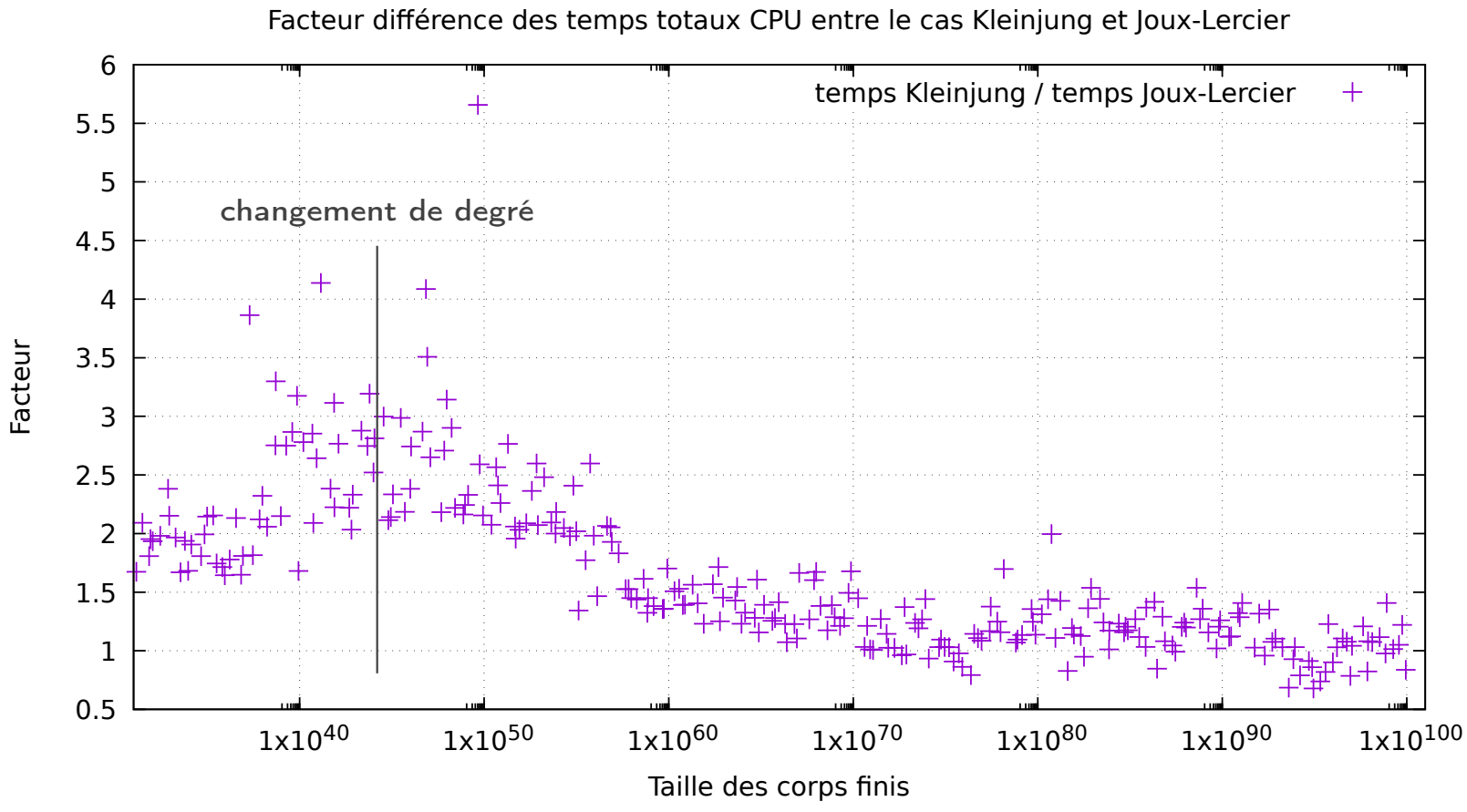
Sélection polynomiale Joux-Lercier et Kleinjung



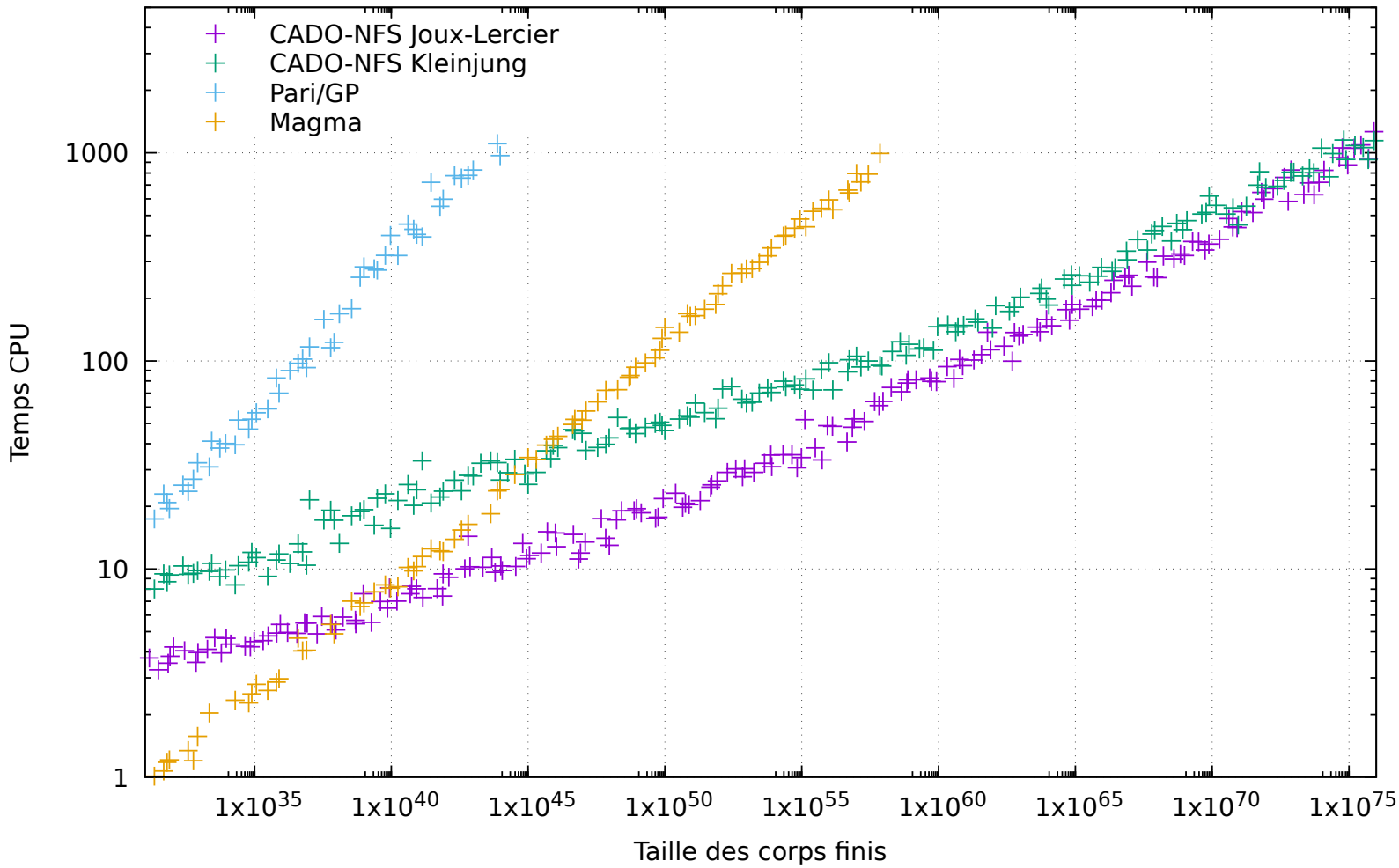
Sélection polynomiale Joux-Lercier et Kleinjung



Sélection polynomiale Joux-Lercier et Kleinjung



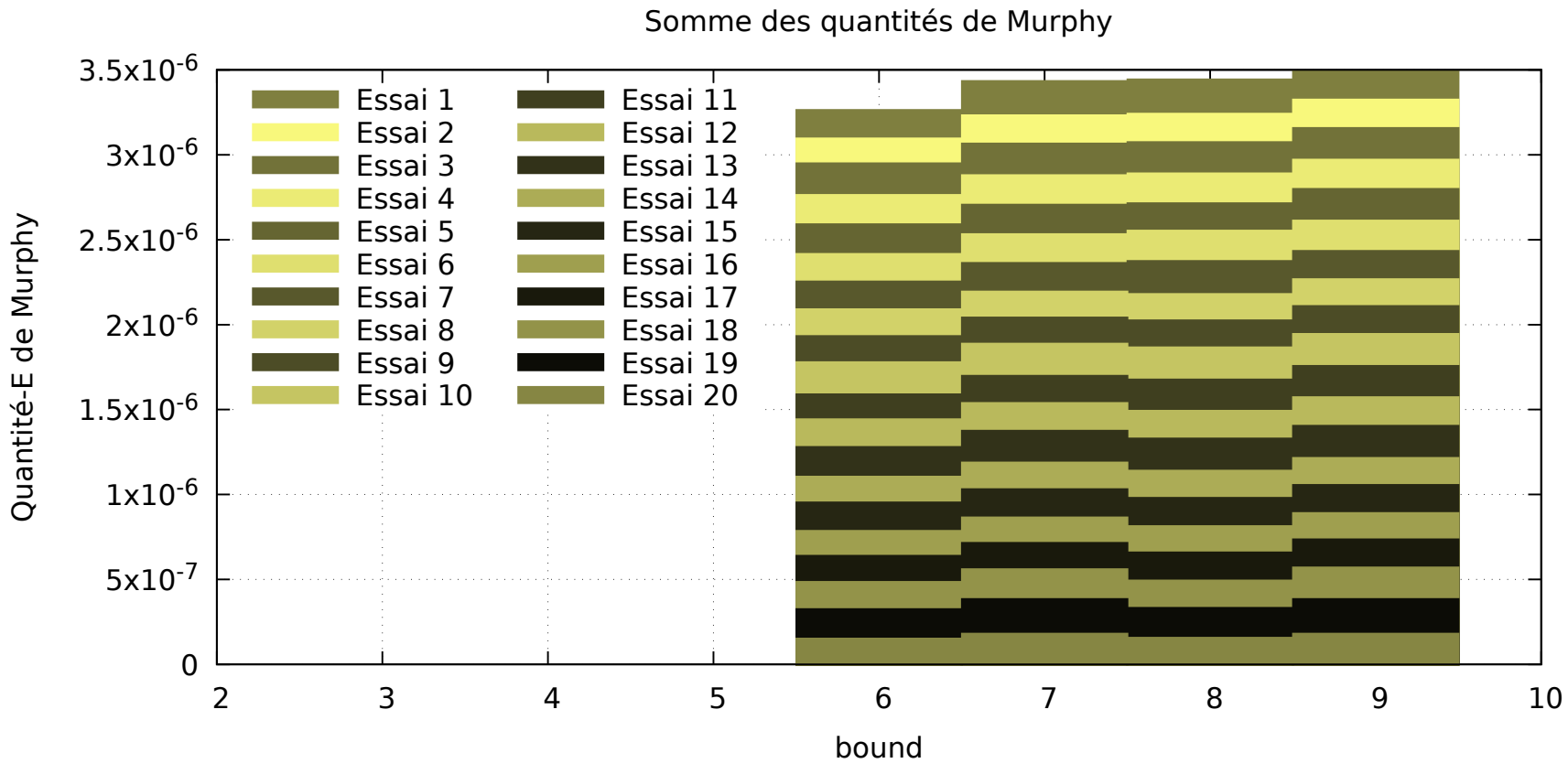
CADO-NFS et d'autres implémentations du logarithme discret



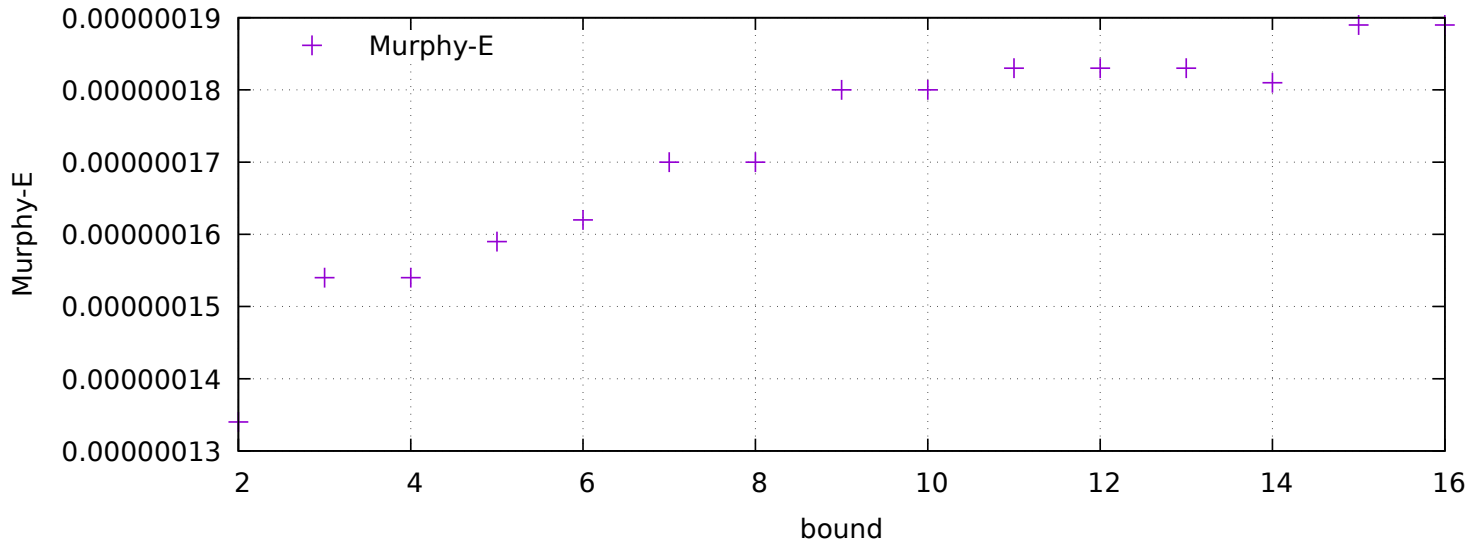
Conclusion

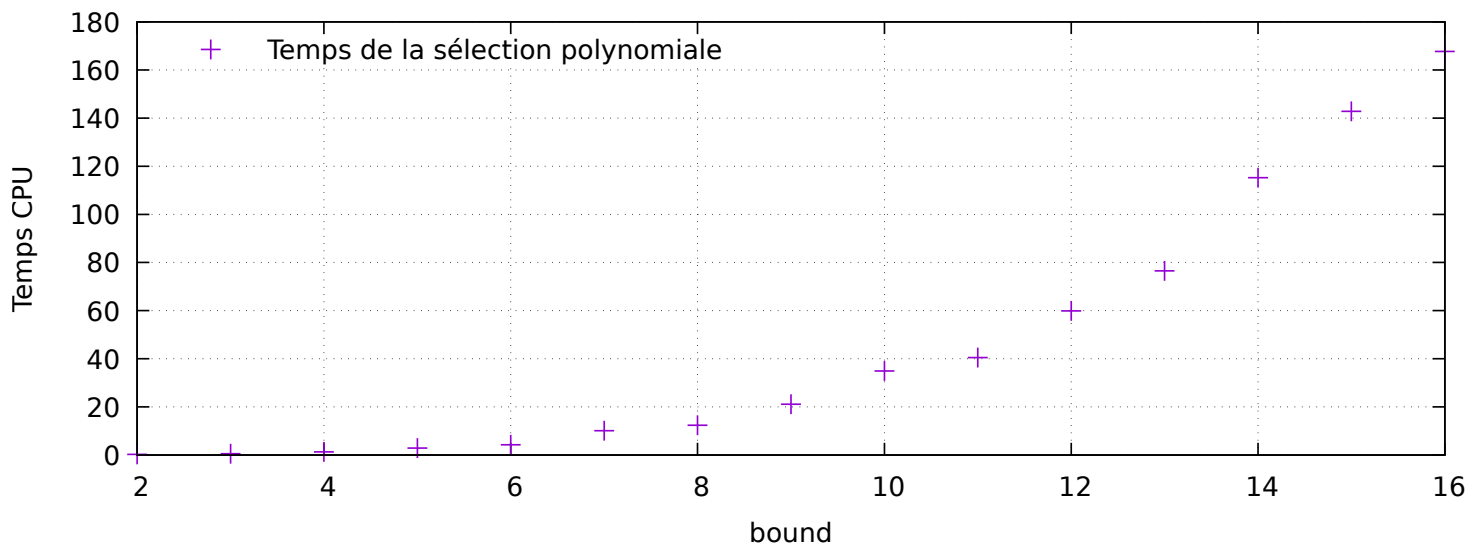
- Paramétrage de CADO-NFS, un problème difficile mais des approches sont possibles.
- Des problèmes négligeables avec les grandes tailles ont leur importance avec les petites tailles.
- La sélection polynomiale Joux-Lercier : une alternative intéressante à l'algorithme de Kleinjung pour le logarithme discret pour les petites tailles ! Toujours vrai pour les grandes tailles ?
- SageMath + CADO-NFS ?

Paramétrage sélection polynomiale Joux-Lercier

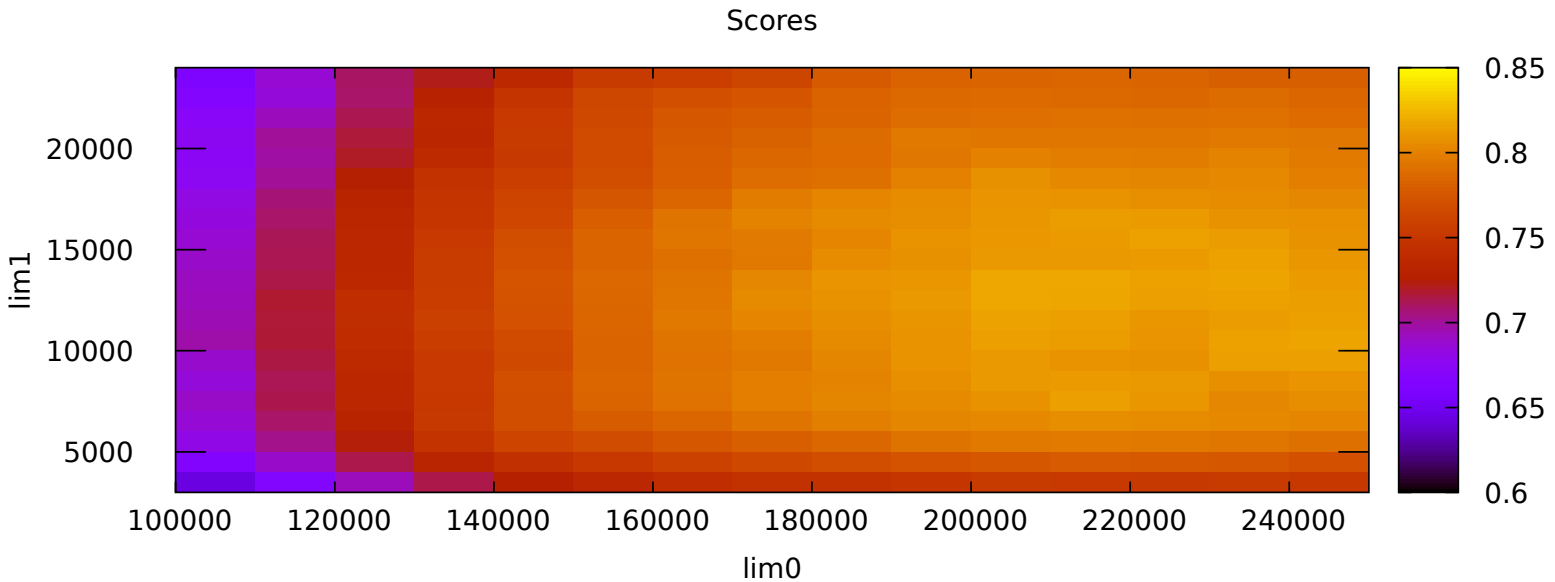


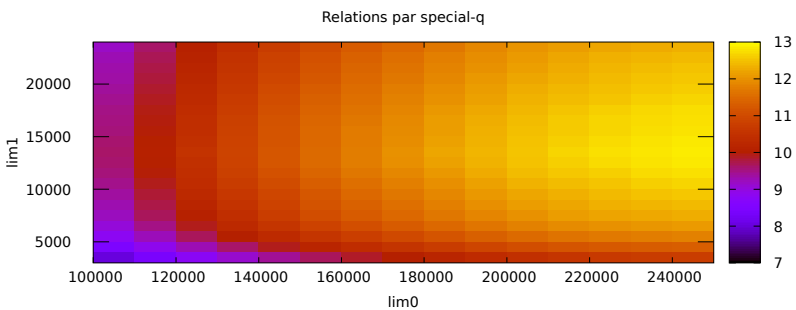
Paramétrage sélection polynomiale Joux-Lercier temps

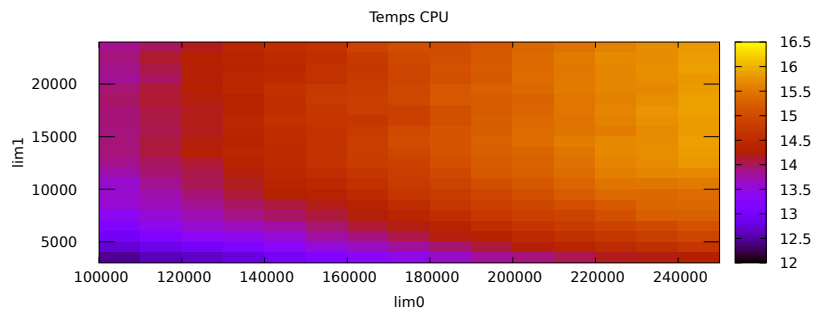




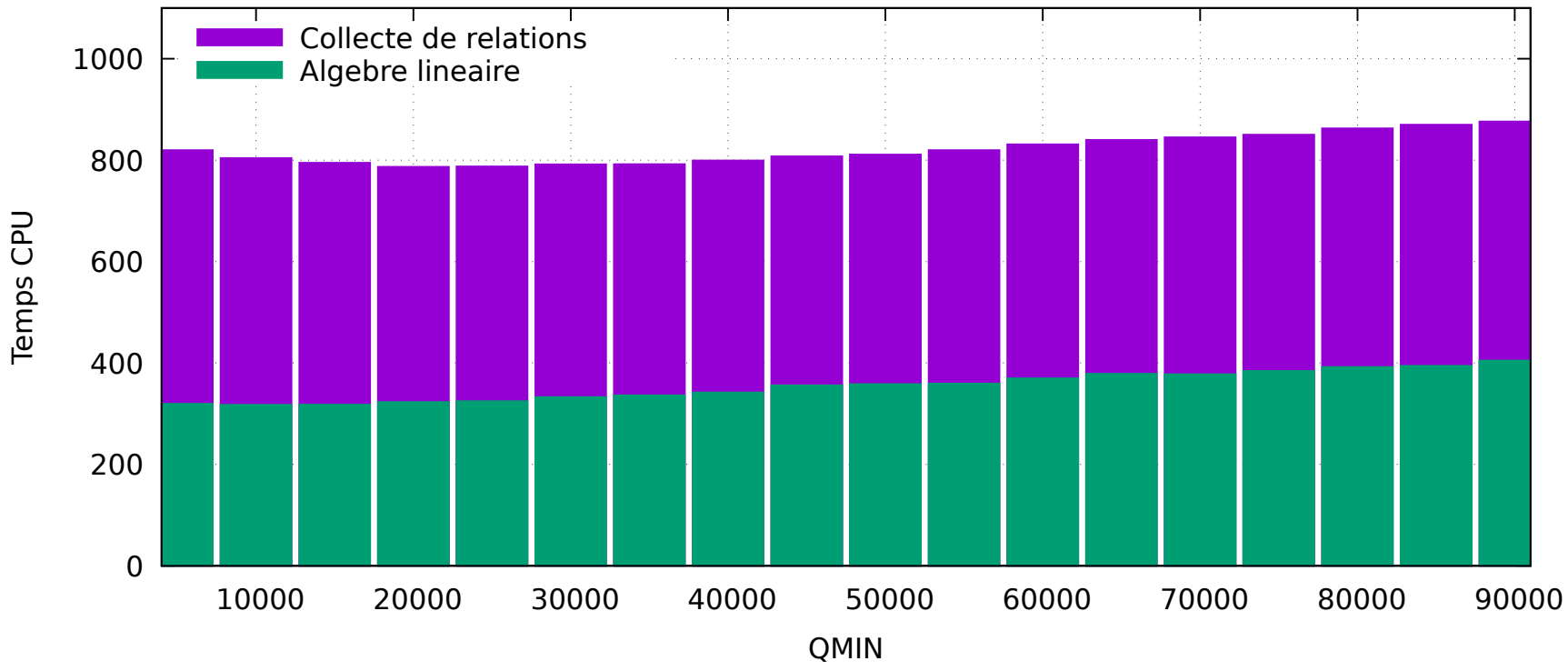
Exemple : paramétrage de la collecte de relations Joux-Lercier



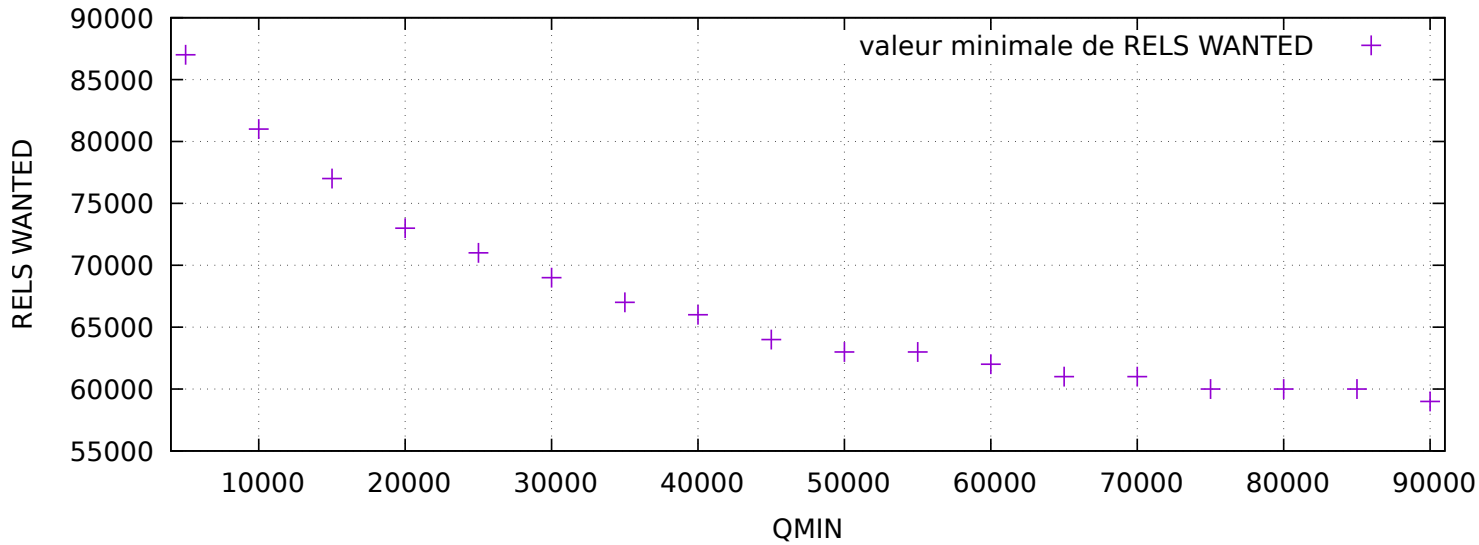


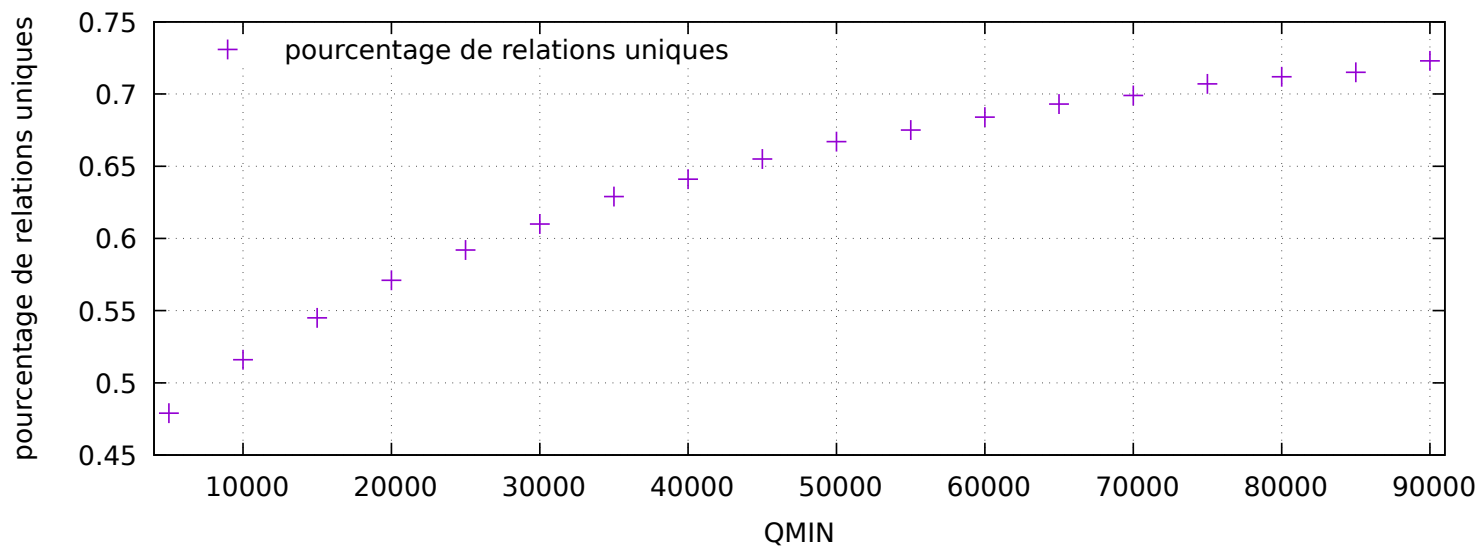


Exemple : Collecte pour l'algèbre linéaire

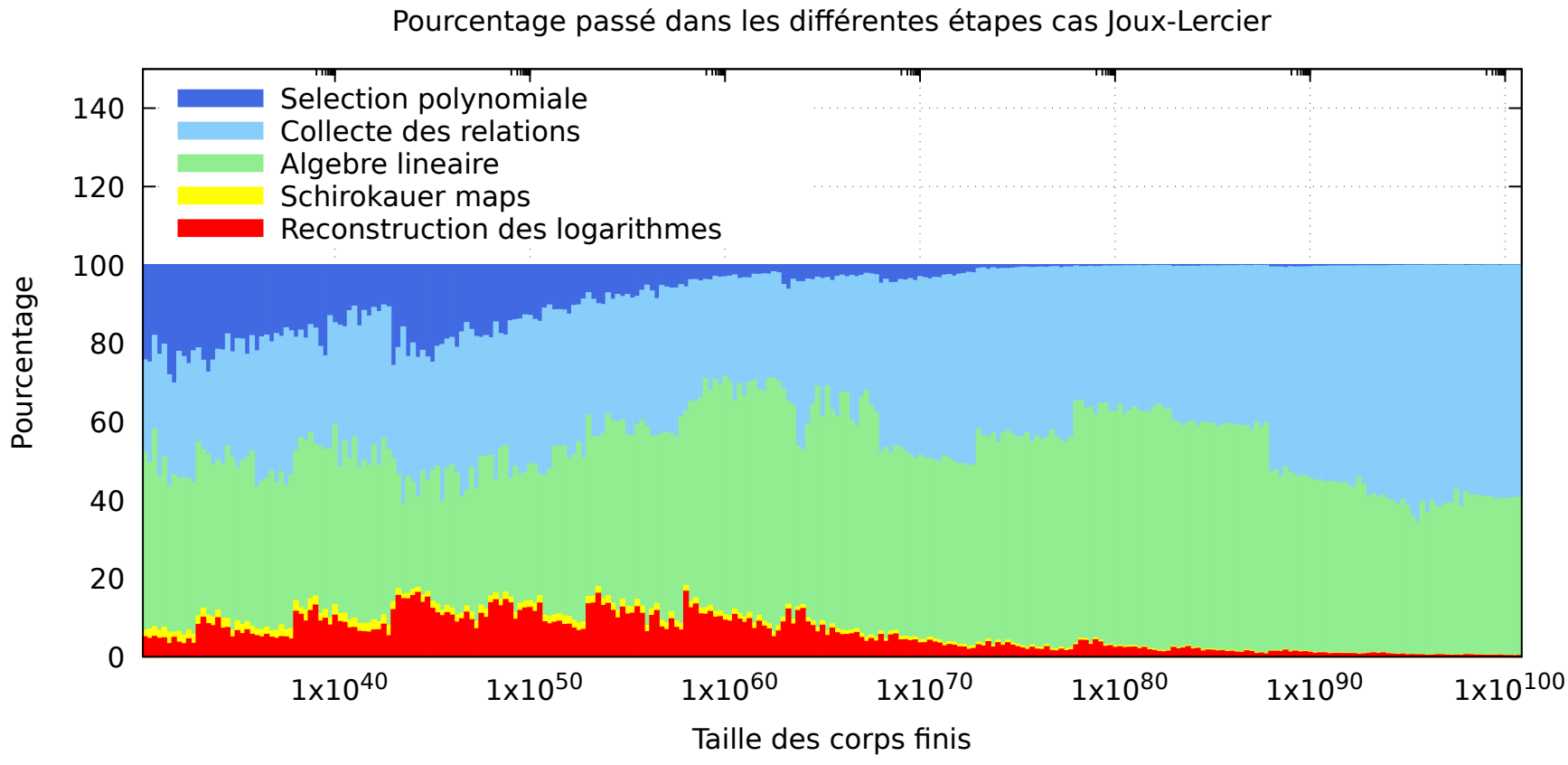


Exemple : Collecte pour l'algèbre linéaire

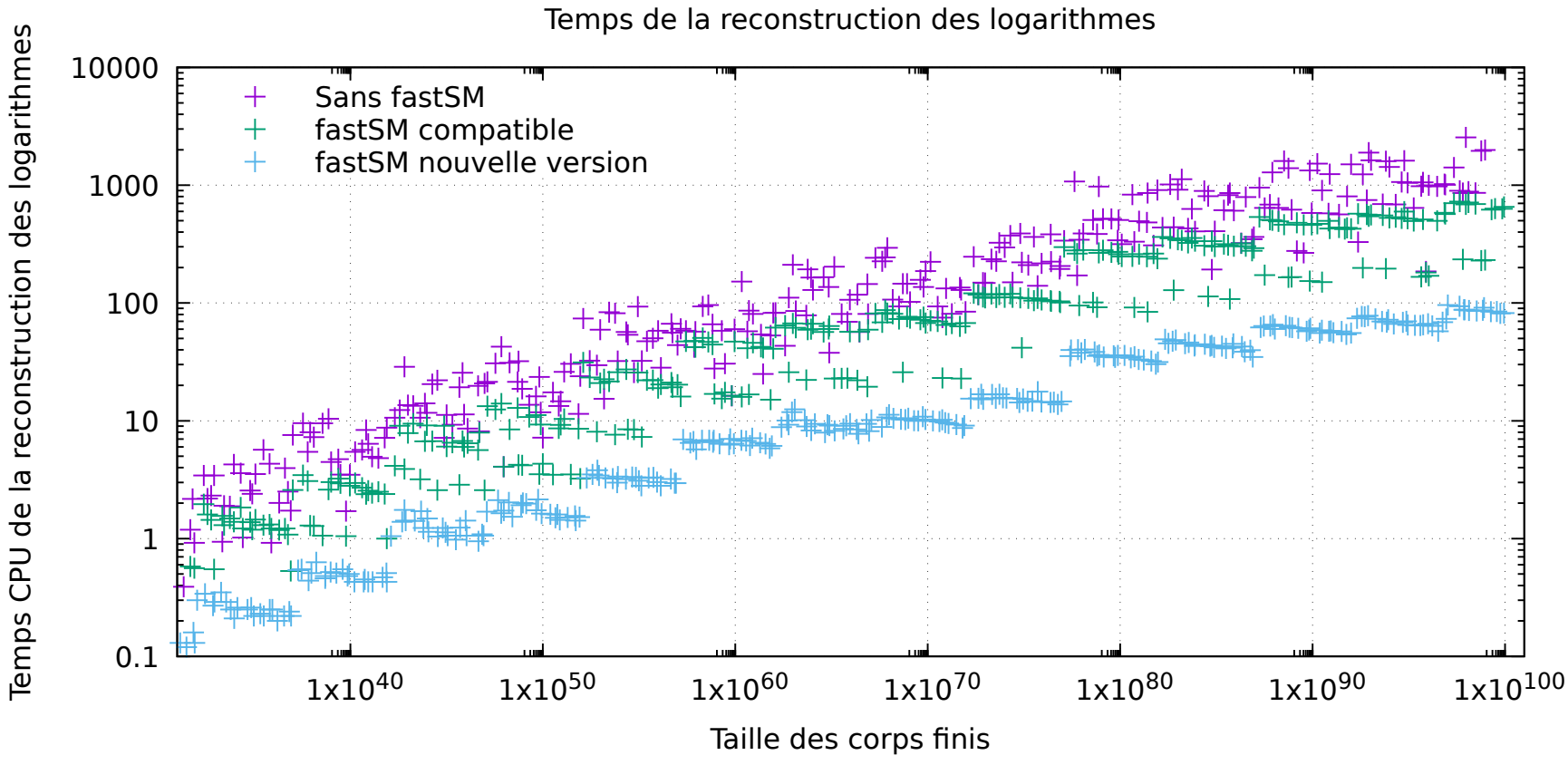




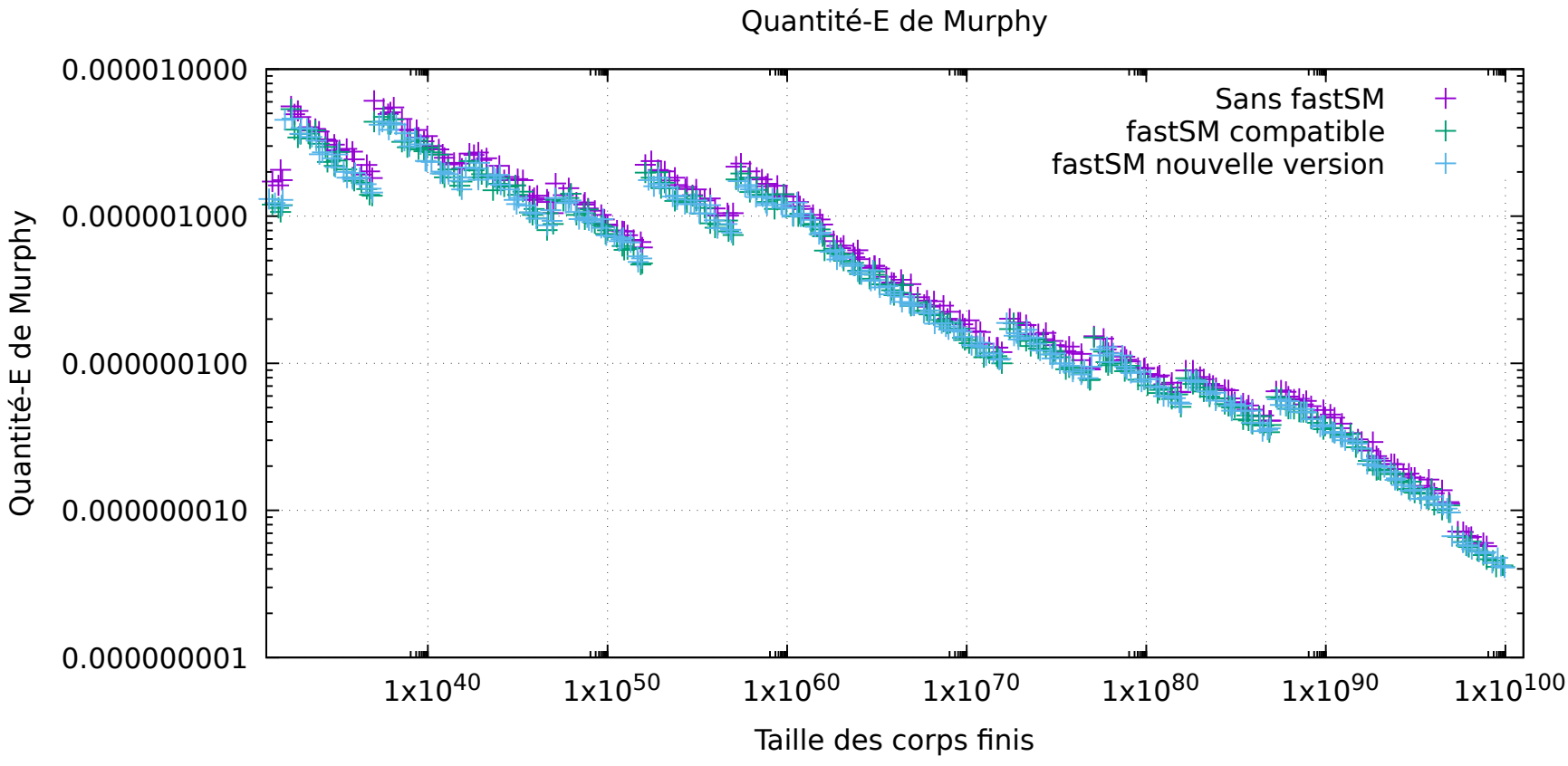
Exemple : Temps et petites tailles



Temps reconstruction des logarithmes



Quantité de Murphy différentes solutions



Comparaison sans biais