

# Benchmarks

Sur une matrice aléatoire de 10x10 de déterminant 11

- Coupe de moitié fonctionne comme attendu
- Coupe par mise à jour améliore agréablement
- L'approximation LLL améliore drastiquement

$B$	$\lambda$	$\tilde{B}$
<div><math display="block">\begin{bmatrix} 10 &amp; 0 &amp; 5 &amp; 5 &amp; 0 &amp; 6 &amp; 7 &amp; 1 &amp; 5 &amp; 6 \\ 11 &amp; 4 &amp; 0 &amp; 11 &amp; 5 &amp; 9 &amp; 2 &amp; 0 &amp; 8 &amp; 4 \\ 1 &amp; 11 &amp; 11 &amp; 12 &amp; 2 &amp; 2 &amp; 3 &amp; 1 &amp; 0 &amp; 2 \\ 4 &amp; 4 &amp; 5 &amp; 7 &amp; 11 &amp; 3 &amp; 4 &amp; 2 &amp; 11 &amp; 7 \\ 3 &amp; 8 &amp; 2 &amp; 7 &amp; 10 &amp; 11 &amp; 11 &amp; 1 &amp; 4 &amp; 4 \\ 7 &amp; 7 &amp; 1 &amp; 5 &amp; 6 &amp; 1 &amp; 6 &amp; 0 &amp; 7 &amp; 1 \\ 10 &amp; 8 &amp; 5 &amp; 8 &amp; 4 &amp; 2 &amp; 12 &amp; 1 &amp; 11 &amp; 8 \\ 0 &amp; 0 &amp; 4 &amp; 10 &amp; 6 &amp; 5 &amp; 9 &amp; 6 &amp; 3 &amp; 1 \\ 10 &amp; 11 &amp; 11 &amp; 2 &amp; 4 &amp; 7 &amp; 9 &amp; 4 &amp; 7 &amp; 1 \\ 8 &amp; 1 &amp; 11 &amp; 12 &amp; 9 &amp; 10 &amp; 0 &amp; 6 &amp; 2 &amp; 7 \end{bmatrix}</math></div>	<div><math display="block">\begin{bmatrix} -1 \\ -1 \\ 1 \\ 1 \\ -2 \\ 5 \\ 1 \\ -2 \\ 0 \\ 5 \end{bmatrix}</math></div>	<div><math display="block">\begin{bmatrix} 1 &amp; 1 &amp; -7 &amp; -4 &amp; -3 &amp; 1 &amp; -2 &amp; -2 &amp; 2 &amp; 1 \\ 1 &amp; 0 &amp; 5 &amp; 7 &amp; 0 &amp; -4 &amp; -3 &amp; 4 &amp; -2 &amp; 7 \\ -1 &amp; 1 &amp; 1 &amp; -1 &amp; 3 &amp; 2 &amp; -1 &amp; 0 &amp; 7 &amp; 6 \\ -1 &amp; 2 &amp; -2 &amp; 3 &amp; -3 &amp; -4 &amp; 3 &amp; -5 &amp; 3 &amp; 1 \\ 2 &amp; 1 &amp; -1 &amp; 2 &amp; 7 &amp; 0 &amp; -1 &amp; 4 &amp; 1 &amp; -5 \\ -5 &amp; 0 &amp; 1 &amp; -4 &amp; 0 &amp; -6 &amp; -1 &amp; 5 &amp; -5 &amp; 4 \\ -1 &amp; 1 &amp; 2 &amp; -1 &amp; 1 &amp; -3 &amp; -8 &amp; -6 &amp; -6 &amp; -2 \\ 2 &amp; 6 &amp; 2 &amp; -4 &amp; 0 &amp; -2 &amp; 3 &amp; -4 &amp; -3 &amp; 3 \\ 0 &amp; 4 &amp; 4 &amp; -4 &amp; -2 &amp; -6 &amp; -1 &amp; 2 &amp; 2 &amp; -1 \\ -5 &amp; 6 &amp; 2 &amp; -2 &amp; 1 &amp; 5 &amp; 6 &amp; 2 &amp; 0 &amp; -2 \end{bmatrix}</math></div>

## Exact SVP

Bench	Temps moyen
Naive	<b>1.7942 s</b>
Half	<b>1.0757 s</b>
Cut	<b>620.28 ms</b>
Half+Cut	<b>371.21 ms</b>
LLL	<b>12.885 ms</b>
All	<b>7.8795 ms</b>

# Benchmarks

Sur une matrice difficile 10x10 de déterminant 28561

- Deux premiers temps estimés à partir d’une seule itération
- Coupe par mise à jour améliore beaucoup
- L’approximation LLL améliore drastiquement

$B$ 
$$\begin{bmatrix} 13 & 0 & 0 & 0 & 4 & -2 & -5 & -5 & -5 & -2 \\ 0 & 13 & 0 & 0 & 1 & 6 & 2 & 0 & 3 & 4 \\ 0 & 0 & 13 & 0 & -3 & -5 & -2 & 6 & 5 & 0 \\ 0 & 0 & 0 & 13 & -3 & -2 & -3 & 6 & -1 & -4 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$\lambda$ 
$$\begin{bmatrix} 0 \\ -1 \\ -1 \\ 0 \\ 0 \\ -1 \\ -1 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

$\tilde{B}$ 
$$\begin{bmatrix} 0 & 0 & 0 & -1 & 1 & 0 & 3 & 1 & 0 & 1 \\ -1 & 0 & 2 & 0 & -1 & 1 & 0 & 1 & 1 & 1 \\ -1 & 1 & 1 & -2 & -1 & 0 & -1 & -2 & -2 & 0 \\ 0 & 1 & 1 & 0 & 1 & 2 & -1 & -2 & -2 & -1 \\ 0 & -1 & 2 & -1 & -1 & 1 & 0 & 1 & -1 & -2 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \\ -1 & -3 & 0 & 0 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 2 & -2 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & 1 & 2 \\ 1 & 0 & 0 & 1 & -1 & -2 & 0 & 0 & 0 & -1 \end{bmatrix}$$

## Exact SVP

Bench	Temps moyen
Naive	~ 16 m
Half	~ 8 m
Cut	3.1994 s
Half+Cut	1.8768 s
LLL	35.651 ms
All	21.415 ms