

Audition CNRS – Concours 06/03

# Réseaux structurés : cryptanalyse et nouvelles constructions

Alice Pellet-Mary

- ▶ Janvier - mars 2020 : chercheure invitée au Simons institute, Berkeley
- ▶ Depuis novembre 2019 : Post-doctorante à KU Leuven, avec Frederik Vercauteren
- ▶ De 2016 à 2019 : Doctorante à l'ENS de Lyon, sous la direction de Damien Stehlé

## Primitives cryptographiques

chiffrement	signature	chiffrement homomorphe	chiffrement fonctionnel	...
-------------	-----------	---------------------------	----------------------------	-----

codes correcteurs	réseaux euclidiens	isogénies
factorisation	logarithme discret	...

## Problèmes algorithmiques (supposés difficiles)

# Problèmes algorithmiques et cryptographie

## Primitives cryptographiques

chiffrement	signature	chiffrement homomorphe	chiffrement fonctionnel	...
-------------	-----------	---------------------------	----------------------------	-----

codes correcteurs	réseaux euclidiens	isogénies
<del>factorisation</del>	<del>logarithme discret</del>	...

Problèmes algorithmiques (supposés difficiles)  
dans un monde quantique

# Problèmes algorithmiques et cryptographie

## Primitives cryptographiques

chiffrement      signature      chiffrement  
   homomorphe      chiffrement  
   fonctionnel      ...

codes correcteurs      réseaux euclidiens      isogénies  
~~factorisation~~      ~~logarithme discret~~      ...

Problèmes algorithmiques (supposés difficiles)  
dans un monde quantique

# Problèmes algorithmiques et cryptographie

## Primitives cryptographiques

chiffrement    signature    chiffrement homomorphe    chiffrement fonctionnel    ...

codes correcteurs    réseaux euclidiens    isogénies  
~~factorisation~~    ~~logarithme discret~~    ...

Problèmes algorithmiques (supposés difficiles)  
dans un monde quantique

# Problèmes algorithmiques et cryptographie

## Primitives cryptographiques

chiffrement    signature    chiffrement homomorphe    chiffrement fonctionnel    ...

codes correcteurs

**réseaux euclidiens**

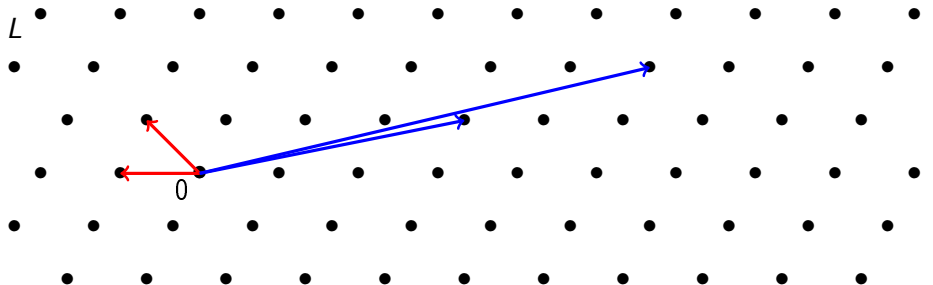
isogénies

~~factorisation~~

~~logarithme discret~~

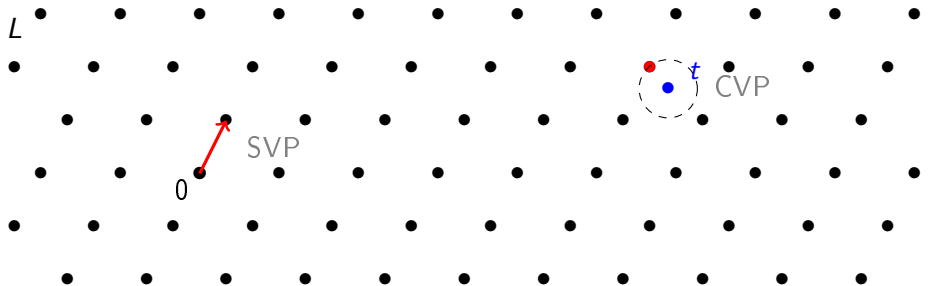
...

Problèmes algorithmiques (supposés difficiles)  
dans un monde quantique



- ▶  $L = \{Bx \mid x \in \mathbb{Z}^n\}$  est un **réseau**
- ▶  $B \in GL_n(\mathbb{R})$  est une **base**
- ▶  $n$  est la **dimension** de  $L$

# Problèmes algorithmiques

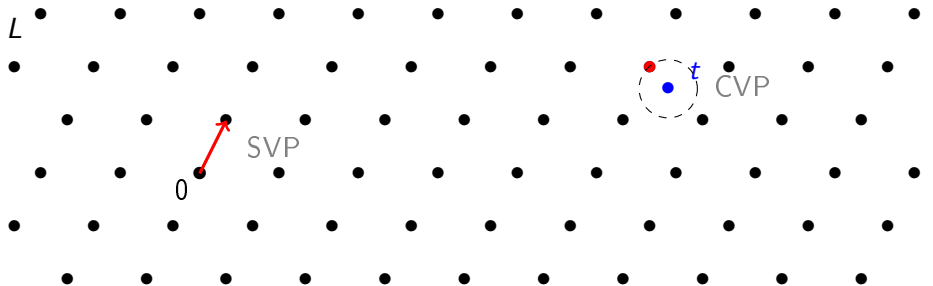


SVP : Shortest Vector Problem

CVP : Closest Vector Problem



# Problèmes algorithmiques



SVP : Shortest Vector Problem

CVP : Closest Vector Problem

Problèmes (supposés) difficiles

- même avec un ordinateur quantique
- même en autorisant une approximation  $\text{poly}(n)$

Réseau module : réseau euclidien + structure algébrique

Réseau module : **réseau euclidien** + structure algébrique

- Réseau de dimension  $n$  sur  $\mathbb{Z}$

Réseau module : réseau euclidien + structure algébrique

- Réseau de dimension  $n = kd$  sur  $\mathbb{Z}$
- Réseau de rang  $k$  sur  $R := \mathbb{Z}[X]/P(X)$ ,  
avec  $P$  irréductible de degré  $d$  (en fait, anneau des entiers  
d'un corps de nombres)

e.g.,  $k = 3$ ,  $d = 256$ ,  $n = 768$

Réseau module : réseau euclidien + structure algébrique

- Réseau de dimension  $n = kd$  sur  $\mathbb{Z}$
- Réseau de rang  $k$  sur  $R := \mathbb{Z}[X]/P(X)$ ,  
avec  $P$  irréductible de degré  $d$  (en fais, anneau des entiers  
d'un corps de nombres)

Avantages/inconvénients :

- + Constructions plus efficaces
- SVP restreint aux réseaux modules pourrait être plus facile

e.g.,  $k = 3$ ,  $d = 256$ ,  $n = 768$

Réseau module : réseau euclidien + structure algébrique

- Réseau de dimension  $n = kd$  sur  $\mathbb{Z}$
- Réseau de rang  $k$  sur  $R := \mathbb{Z}[X]/P(X)$ ,  
avec  $P$  irréductible de degré  $d$  (en fais, anneau des entiers  
d'un corps de nombres)

Avantages/inconvénients :

- + Constructions plus efficaces
- SVP restreint aux réseaux modules pourrait être plus facile  
(pas pour l'instant)

e.g.,  $k = 3$ ,  $d = 256$ ,  $n = 768$

Un de mes résultats : LLL pour les réseaux modules

LLL sur  $\mathbb{Z}$

Résout SVP

**Temps** :  $\text{poly}(n)$

**Approx** :  $2^n$

---

[LLL82] A. K. Lenstra, H. W. Lenstra, L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*.



## LLL sur $\mathbb{Z}$

Résout SVP

**Temps** :  $\text{poly}(n)$

**Approx** :  $2^n$

## LLL sur $R$ ?

Résoudrait SVP dans les modules de rang  $k$

**Temps** :  $\text{poly}(d, k)$

**Approx** :  $\text{poly}(d) \cdot 2^k$

---

[LLL82] A. K. Lenstra, H. W. Lenstra, L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*.

## LLL sur $\mathbb{Z}$

Résout SVP

**Temps** :  $\text{poly}(n)$

**Approx** :  $2^n$

## LLL sur $R$ ?

Résoudrait SVP dans les modules de rang  $k$

**Temps** :  $\text{poly}(d, k)$

**Approx** :  $\text{poly}(d) \cdot 2^k$

► Attaquerait 11 des 26 candidats du NIST<sup>1</sup>

---

[LLL82] A. K. Lenstra, H. W. Lenstra, L. Lovász. Factoring polynomials with rational coefficients. Mathematische Annalen.

<sup>1</sup> Processus de standardisation post-quantique débuté en 2017

## LLL sur $\mathbb{Z}$

Résout SVP

**Temps** :  $\text{poly}(n)$

**Approx** :  $2^n$

## LLL sur $R$ ?

Résoudrait SVP dans les modules de rang  $k$

**Temps** :  $\text{poly}(d, k)$

**Approx** :  $\text{poly}(d) \cdot 2^k$

- ▶ Attaquerait 11 des 26 candidats du NIST<sup>1</sup>

Problème ouvert depuis plus de 20 ans [Nap96, FP96, KL17]  
⇒ jusqu'à présent : généralisation seulement si  $d \leq 4$

---

[Nap96] H. Napias. A generalization of the LLL-algorithm over Euclidean rings or orders. Journal de théorie des nombres de Bordeaux.

[FP96] C. Fieker, M. E. Pohst. Lattices over number fields. ANTS.

[KL17] T. Kim, C. Lee. Lattice reductions over euclidean rings with applications to cryptanalysis. IMACC.

## *An LLL Algorithm for Module Lattices*

avec C. Lee, D. Stehlé et A. Wallet, Asiacrypt 2019.

## *An LLL Algorithm for Module Lattices*

avec C. Lee, D. Stehlé et A. Wallet, Asiacrypt 2019.

Algorithme LLL pour n'importe quel  $P$  de degré  $d$  :

- Approx :  $\text{quasipoly}(d)^k$  (pour les cyclotomiques)
- Temps :  $\text{poly}(k, d)$  à condition d'avoir un oracle résolvant CVP dans un réseau  $L = L(R)$  fixé

## *An LLL Algorithm for Module Lattices*

avec C. Lee, D. Stehlé et A. Wallet, Asiacrypt 2019.

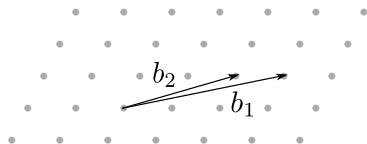
Algorithme LLL pour n'importe quel  $P$  de degré  $d$  :

- Approx :  $\text{quasipoly}(d)^k$  (pour les cyclotomiques)
- Temps :  $\text{poly}(k, d)$  à condition d'avoir un oracle résolvant CVP dans un réseau  $L = L(R)$  fixé

À retenir :

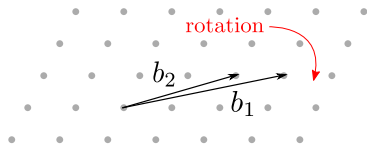
- ★ Premier algorithme LLL pour tous les polynômes  $P$
- ★ Bon formalisme pour passer de  $\mathbb{Z}$  à  $R$

# LLL sur $\mathbb{Z}$ en dimension 2 (Lagrange-Gauss)



$$M = \begin{pmatrix} 10 & 7 \\ 2 & 2 \end{pmatrix}$$

# LLL sur $\mathbb{Z}$ en dimension 2 (Lagrange-Gauss)



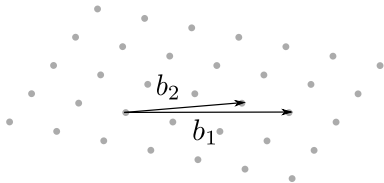
rotation

$$M = \begin{pmatrix} 10 & 7 \\ 2 & 2 \end{pmatrix}$$

factorisation QR

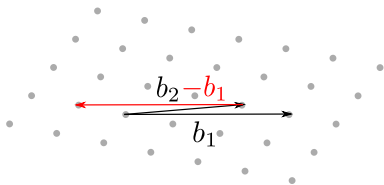


# LLL sur $\mathbb{Z}$ en dimension 2 (Lagrange-Gauss)



$$M = \begin{pmatrix} 10,2 & 7,3 \\ 0 & 0,6 \end{pmatrix}$$

# LLL sur $\mathbb{Z}$ en dimension 2 (Lagrange-Gauss)

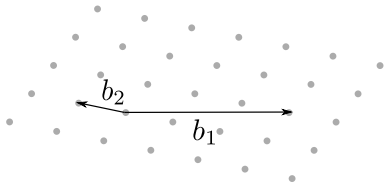


réduire  $b_2$  avec  $b_1$

$$M = \begin{pmatrix} 10,2 & 7,3 \\ 0 & 0,6 \end{pmatrix}$$

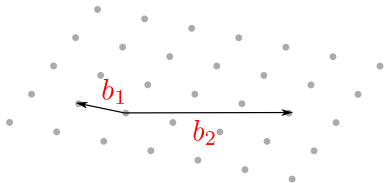
“division euclidienne centrée”  
(sur  $\mathbb{R}$ ) de 7,3 par 10,2

# LLL sur $\mathbb{Z}$ en dimension 2 (Lagrange-Gauss)



$$M = \begin{pmatrix} 10,2 & -2,9 \\ 0 & 0,6 \end{pmatrix}$$

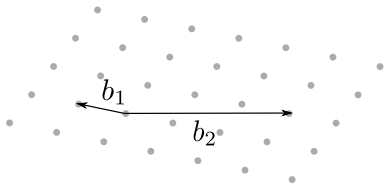
# LLL sur $\mathbb{Z}$ en dimension 2 (Lagrange-Gauss)



$$M = \begin{pmatrix} -2,9 & 10,2 \\ 0,6 & 0 \end{pmatrix}$$

échanger  $b_1$  et  $b_2$

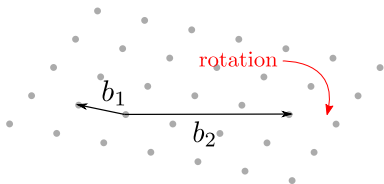
# LLL sur $\mathbb{Z}$ en dimension 2 (Lagrange-Gauss)



recommencer

$$M = \begin{pmatrix} -2,9 & 10,2 \\ 0,6 & 0 \end{pmatrix}$$

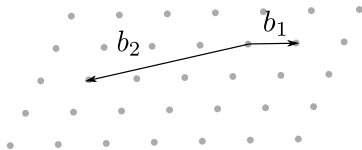
# LLL sur $\mathbb{Z}$ en dimension 2 (Lagrange-Gauss)



rotation

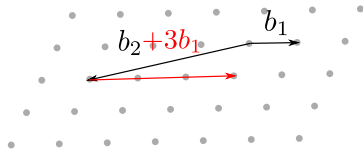
$$M = \begin{pmatrix} -2,9 & 10,2 \\ 0,6 & 0 \end{pmatrix}$$

# LLL sur $\mathbb{Z}$ en dimension 2 (Lagrange-Gauss)



$$M = \begin{pmatrix} 3 & -10 \\ 0 & -2 \end{pmatrix}$$

# LLL sur $\mathbb{Z}$ en dimension 2 (Lagrange-Gauss)



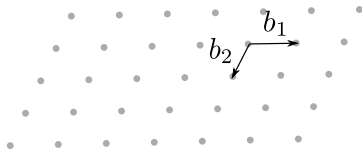
réduire  $b_2$  avec  $b_1$

$$M = \begin{pmatrix} 3 & -10 \\ 0 & -2 \end{pmatrix}$$

“division euclidienne centrée”  
(sur  $\mathbb{R}$ ) de  $-10$  par  $3$

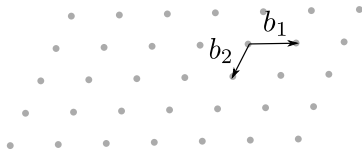


# LLL sur $\mathbb{Z}$ en dimension 2 (Lagrange-Gauss)



$$M = \begin{pmatrix} 3 & -1 \\ 0 & -2 \end{pmatrix}$$

# LLL sur $\mathbb{Z}$ en dimension 2 (Lagrange-Gauss)



$$M = \begin{pmatrix} 3 & -1 \\ 0 & -2 \end{pmatrix}$$

## Ingrédient principal

La division euclidienne (et la factorisation QR)

# Une pseudo-division euclidienne sur $R$

Entrée :  $a, b \in R$

Sortie :  $r \in R$  t.q.

$$\|b/a - r\| \leq 1/2$$

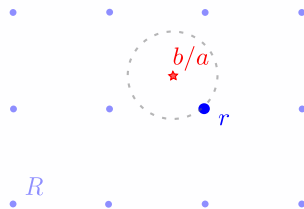
Division euclidienne  $\sim$  CVP dans  $R$

# Une pseudo-division euclidienne sur $R$

Entrée :  $a, b \in R$

Sortie :  $r \in R$  t.q.

$$\|b/a - r\| \leq 1/2$$



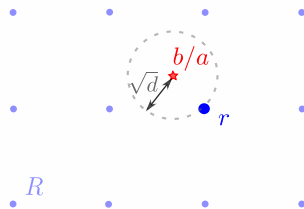
Division euclidienne  $\sim$  CVP dans  $R$

# Une pseudo-division euclidienne sur $R$

Entrée :  $a, b \in R$

Sortie :  $r \in R$  t.q.

$$\|b/a - r\| \leq 1/2$$



Division euclidienne  $\sim$  CVP dans  $R$

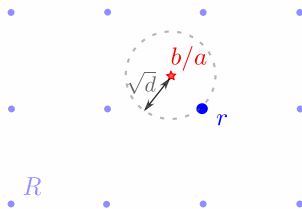
# Une pseudo-division euclidienne sur $R$

Entrée :  $a, b \in R$

Sortie :  $u, v \in R$  t.q.

$$\|b/a - u/v\| \leq 1/\text{poly}(d)$$

$$\text{et } \|v\| \leq \text{poly}(d)$$



Division euclidienne  $\sim$  CVP dans  $R$

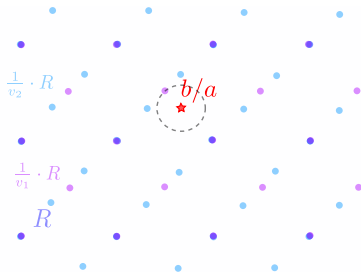
# Une pseudo-division euclidienne sur $R$

Entrée :  $a, b \in R$

Sortie :  $u, v \in R$  t.q.

$$\|b/a - u/v\| \leq 1/\text{poly}(d)$$

$$\text{et } \|v\| \leq \text{poly}(d)$$



Division euclidienne  $\sim$  CVP dans  $R$

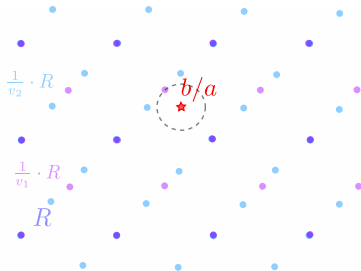
# Une pseudo-division euclidienne sur $R$

Entrée :  $a, b \in R$

Sortie :  $u, v \in R$  t.q.

$$\|b/a - u/v\| \leq 1/\text{poly}(d)$$

$$\text{et } \|v\| \leq \text{poly}(d)$$



Comment calculer  $u$  et  $v$  ?



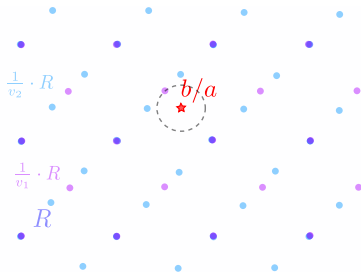
# Une pseudo-division euclidienne sur $R$

Entrée :  $a, b \in R$

Sortie :  $u, v \in R$  t.q.

$$\|b/a - u/v\| \leq 1/\text{poly}(d)$$

$$\text{et } \|v\| \leq \text{poly}(d)$$



Comment calculer  $u$  et  $v$  ?

Techniques : • chercher  $u/v = \prod_i x_i^{\alpha_i}$  avec  $x_i \in R$  fixés et  $\alpha_i \in \mathbb{Z}$

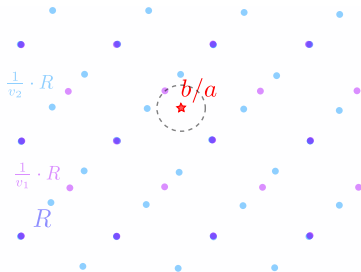
# Une pseudo-division euclidienne sur $R$

Entrée :  $a, b \in R$

Sortie :  $u, v \in R$  t.q.

$$\|b/a - u/v\| \leq 1/\text{poly}(d)$$

$$\text{et } \|v\| \leq \text{poly}(d)$$



Comment calculer  $u$  et  $v$  ?

- Techniques :
- chercher  $u/v = \prod_i x_i^{\alpha_i}$  avec  $x_i \in R$  fixés et  $\alpha_i \in \mathbb{Z}$
  - prendre le log
    - ▶  $\log(u/v) \in L := \{\sum_i \alpha_i \log(x_i), \alpha_i \in \mathbb{Z}\}$

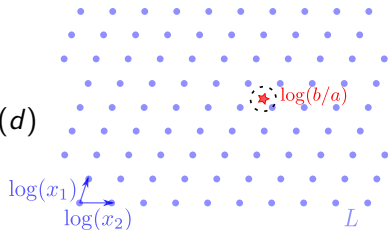
# Une pseudo-division euclidienne sur $R$

Entrée :  $a, b \in R$

Sortie :  $\log(u/v) \in L$  t.q.

$$\|\log(b/a) - \log(u/v)\| \leq 1/\text{poly}(d)$$

$$\text{et } \|\log(v)\| \leq O(\log d)$$



Comment calculer  $u$  et  $v$  ?

- Techniques :
- chercher  $u/v = \prod_i x_i^{\alpha_i}$  avec  $x_i \in R$  fixés et  $\alpha_i \in \mathbb{Z}$
  - prendre le log
    - ▶  $\log(u/v) \in L := \{\sum_i \alpha_i \log(x_i), \alpha_i \in \mathbb{Z}\}$

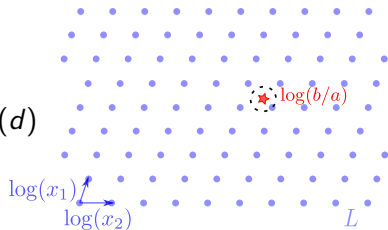
# Une pseudo-division euclidienne sur $R$

Entrée :  $a, b \in R$

Sortie :  $\log(u/v) \in L$  t.q.

$$\|\log(b/a) - \log(u/v)\| \leq 1/\text{poly}(d)$$

$$\text{et } \|\log(v)\| \leq O(\log d)$$



Pseudo-division euclidienne  $\sim$  CVP dans  $L$

CVP dans  $L \Rightarrow$  pseudo-division euclidienne dans  $R$   
 $\Rightarrow$  LLL pour les  $R$ -modules de rang 2

CVP dans  $L \Rightarrow$  pseudo-division euclidienne dans  $R$   
 $\Rightarrow$  LLL pour les  $R$ -modules de rang 2

Sous le tapis :

- le 'vrai'  $L$  contient les unités et le groupe des classes de  $R$

CVP dans  $L \Rightarrow$  pseudo-division euclidienne dans  $R$   
 $\Rightarrow$  LLL pour les  $R$ -modules de rang 2

Sous le tapis :

- le 'vrai'  $L$  contient les unités et le groupe des classes de  $R$
- heuristiques sur la densité de  $L$

CVP dans  $L \Rightarrow$  pseudo-division euclidienne dans  $R$   
 $\Rightarrow$  LLL pour les  $R$ -modules de rang  $k$

Sous le tapis :

- le 'vrai'  $L$  contient les unités et le groupe des classes de  $R$
- heuristiques sur la densité de  $L$
- généralisation au rang  $k$



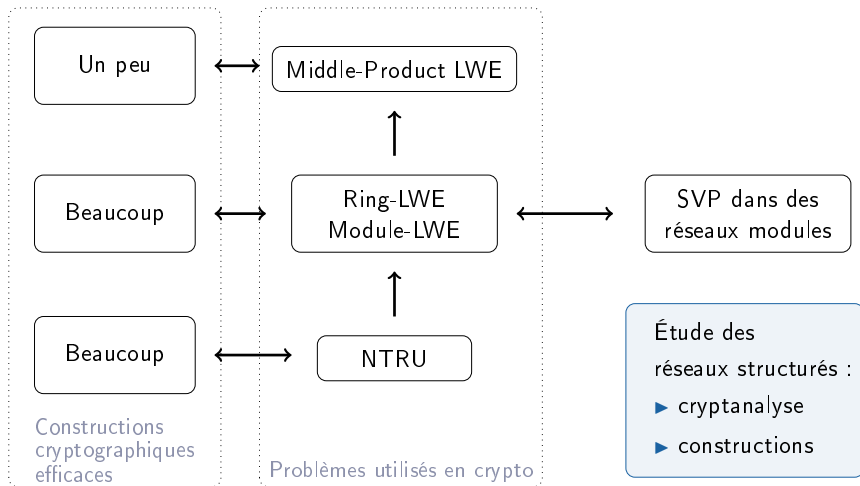
CVP dans  $L \Rightarrow$  pseudo-division euclidienne dans  $R$   
 $\Rightarrow$  LLL pour les  $R$ -modules de rang  $k$

Sous le tapis :

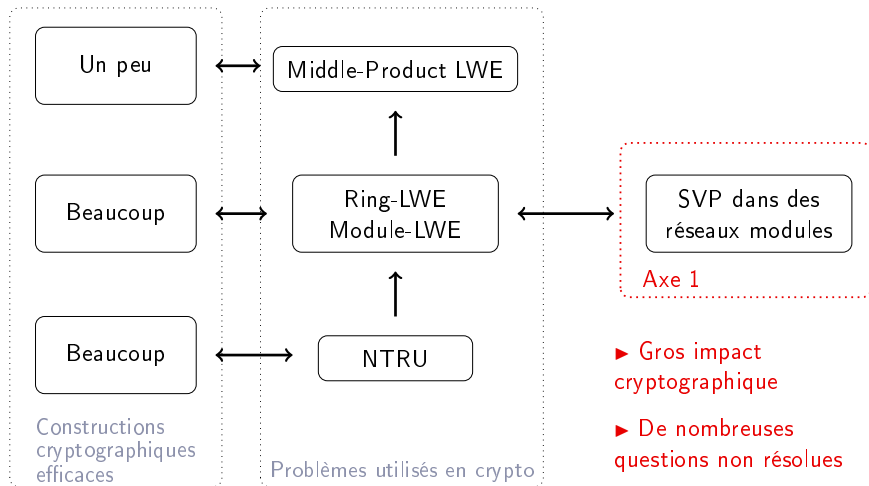
- le 'vrai'  $L$  contient les unités et le groupe des classes de  $R$
- heuristiques sur la densité de  $L$
- généralisation au rang  $k$

Une seule chose manquante : un algorithme efficace pour CVP dans  $L$

## Projet de recherche



# 1. SVP dans les réseaux modules



# 1. SVP dans les réseaux modules

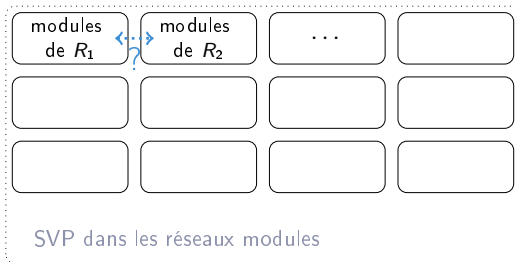
Une question possible :

modules de $R_1$	modules de $R_2$	...	

SVP dans les réseaux modules

# 1. SVP dans les réseaux modules

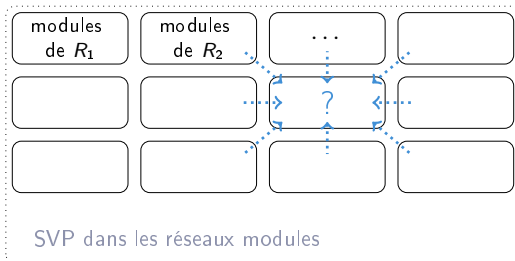
Une question possible :



- Deux anneaux avec la même géométrie ?

# 1. SVP dans les réseaux modules

Une question possible :



- Deux anneaux avec la même géométrie ?
- Un anneau qui en domine d'autres ?

# 1. SVP dans les réseaux modules

Une autre question possible :

[Sch08] : Module de rang 1  $\sim$  classe du groupe des classes d'Arakelov

- ▶ “bon” formalisme pour les modules de rang 1
- ▶ réduction ‘pire cas / moyen cas’ pour les modules de rang 1  
(collaboration avec K. de Boer, L. Ducas et B. Wesolowski)



# 1. SVP dans les réseaux modules

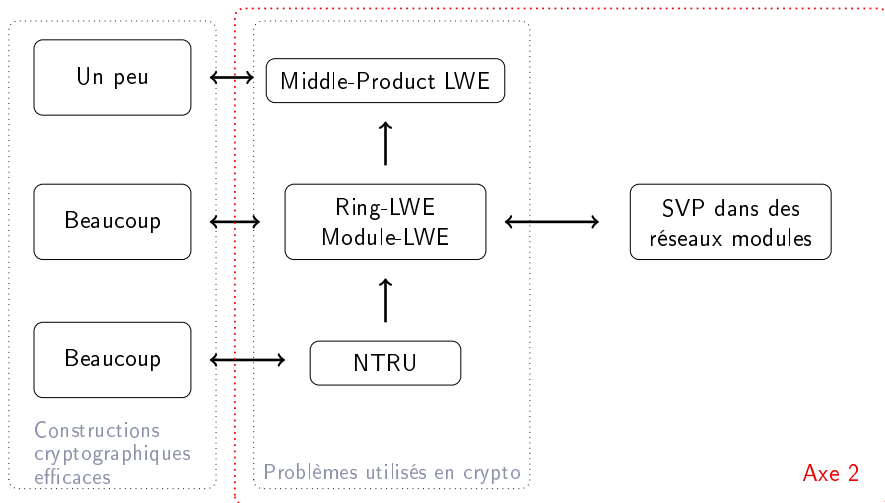
Une autre question possible :

[Sch08] : Module de rang 1  $\sim$  classe du groupe des classes d'Arakelov

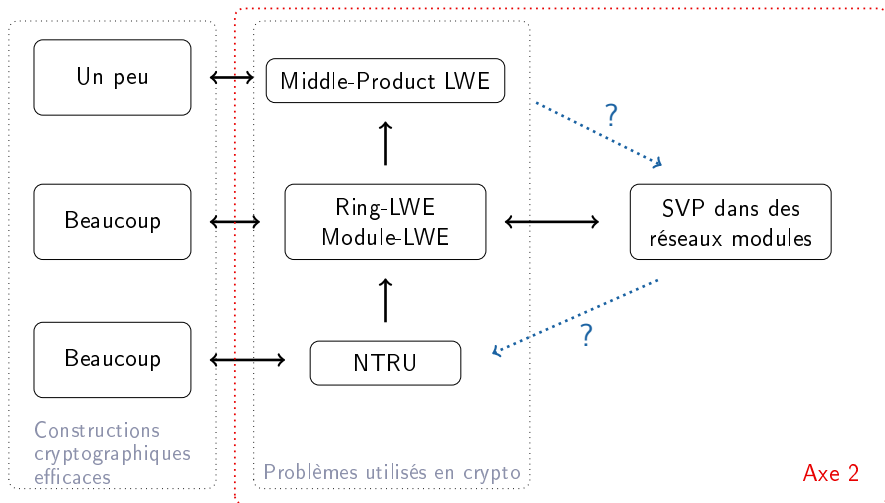
- ▶ “bon” formalisme pour les modules de rang 1
- ▶ réduction ‘pire cas / moyen cas’ pour les modules de rang 1  
(collaboration avec K. de Boer, L. Ducas et B. Wesolowski)

Existe-t-il une généralisation au rang  $k \geq 2$  ?

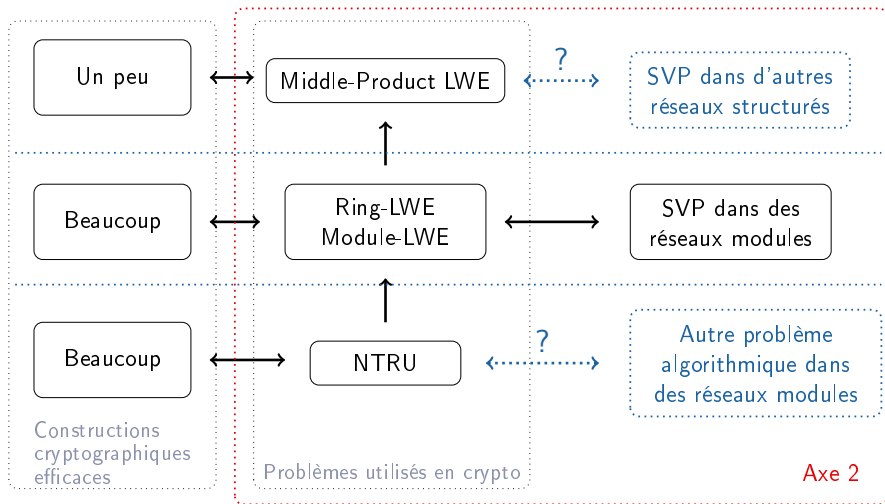
## 2. Lien avec d'autres problèmes algorithmiques



## 2. Lien avec d'autres problèmes algorithmiques

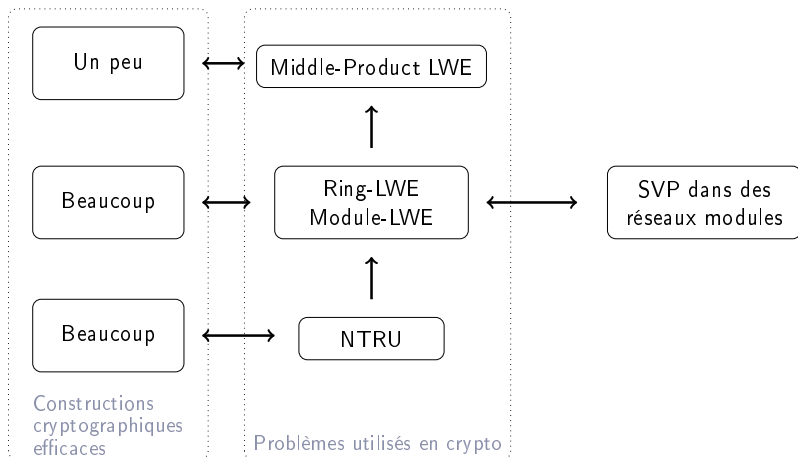


## 2. Lien avec d'autres problèmes algorithmiques

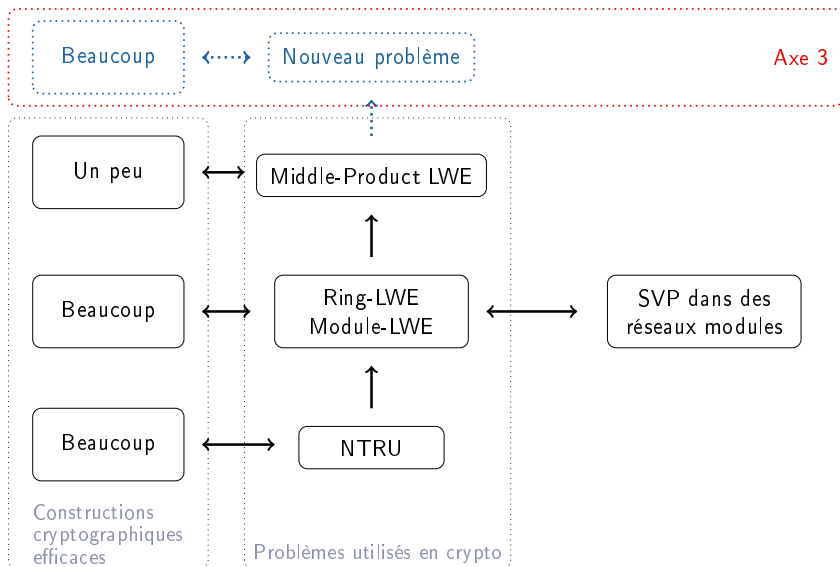


### 3. Nouvelles constructions

Axe 3



### 3. Nouvelles constructions



## Mises à jour :

- Un article accepté à Eurocrypt 2020
  - ▶ avec S. Agrawal (IIT Madras, Chennai)
- Un article accepté à Crypto 2020
  - ▶ avec K. de Boer, L. Ducas et B. Wesolowski (CWI, Amsterdam + CNRS)

## Intégrations possibles :

- équipe de théorie des nombres de l'IMB, Bordeaux
- équipe géométrie et algèbre effective de l'IRMAR, Rennes
- institut Fourier, Grenoble