Modèle générique pour les groupes avec couplage et problèmes bilinéaires

David Lubicz, Thomas Sirvent
david.lubicz@math.univ-rennes1.fr,
thomas.sirvent@m4x.org

CELAR – IRMAR

Plan de l'exposé

Motivation

Les groupes avec couplage sont de plus en plus utilisés dans les protocoles cryptographiques récents :

- Diffie-Hellman à 3 utilisateurs (Joux, ANTS'2000),
- chiffrement basé sur l'identité (Boneh - Franklin, Crypto'2001),
- signature courte (Boneh - Lynn - Shacham, Asiacrypt'2001).

La sécurité de ces protocoles repose sur des hypothèses, pour lesquelles on n'a que des présomptions de robustesse. Est-il possible de fournir un argument positif de sécurité ?

L'utilisation de "groupes génériques" représente un contexte plus faible. On peut espérer prouver ces hypothèses dans ce cadre-là.

Motivation

Les groupes avec couplage sont de plus en plus utilisés dans les protocoles cryptographiques récents :

- Diffie-Hellman à 3 utilisateurs (Joux, ANTS'2000),
- chiffrement basé sur l'identité (Boneh - Franklin, Crypto'2001),
- signature courte (Boneh - Lynn - Shacham, Asiacrypt'2001).

La sécurité de ces protocoles repose sur des hypothèses, pour lesquelles on n'a que des présomptions de robustesse. Est-il possible de fournir un argument positif de sécurité ?

L'utilisation de "groupes génériques" représente un contexte plus faible. On peut espérer prouver ces hypothèses dans ce cadre-là.

Motivation

Les groupes avec couplage sont de plus en plus utilisés dans les protocoles cryptographiques récents :

- Diffie-Hellman à 3 utilisateurs (Joux, ANTS'2000),
- chiffrement basé sur l'identité (Boneh - Franklin, Crypto'2001),
- signature courte (Boneh - Lynn - Shacham, Asiacrypt'2001).

La sécurité de ces protocoles repose sur des hypothèses, pour lesquelles on n'a que des présomptions de robustesse. Est-il possible de fournir un argument positif de sécurité ?

L'utilisation de "groupes génériques" représente un contexte plus faible. On peut espérer prouver ces hypothèses dans ce cadre-là.

Motivation

Les groupes avec couplage sont de plus en plus utilisés dans les protocoles cryptographiques récents :

- Diffie-Hellman à 3 utilisateurs (Joux, ANTS'2000),
- chiffrement basé sur l'identité (Boneh - Franklin, Crypto'2001),
- signature courte (Boneh - Lynn - Shacham, Asiacrypt'2001).

La sécurité de ces protocoles repose sur des hypothèses, pour lesquelles on n'a que des présomptions de robustesse. Est-il possible de fournir un argument positif de sécurité ?

L'utilisation de "groupes génériques" représente un contexte plus faible. On peut espérer prouver ces hypothèses dans ce cadre-là.

Motivation

Les groupes avec couplage sont de plus en plus utilisés dans les protocoles cryptographiques récents :

- Diffie-Hellman à 3 utilisateurs (Joux, ANTS'2000),
- chiffrement basé sur l'identité (Boneh - Franklin, Crypto'2001),
- signature courte (Boneh - Lynn - Shacham, Asiacrypt'2001).

La sécurité de ces protocoles repose sur des hypothèses, pour lesquelles on n'a que des présomptions de robustesse. Est-il possible de fournir un argument positif de sécurité ?

L'utilisation de "groupes génériques" représente un contexte plus faible. On peut espérer prouver ces hypothèses dans ce cadre-là.

Plan de l'exposé

Des algorithmes génériques aux groupes génériques

Un algorithme générique (Shoup, Eurocrypt'1997) est une succession de requêtes (addition ou opposé) dans un groupe. Dans un sens plus large, on peut vouloir autoriser des tests, des boucles, mais rien ne doit permettre à l'algorithme de "prédire" des réponses à certaines requêtes.

La probabilité de succès d'un algorithme générique, sur un problème donné, est déterminée par la proportion de cas favorables dans l'ensemble des groupes compatibles avec les réponses obtenues par l'algorithme.

Le groupe générique est un groupe dont la loi est tirée aléatoirement de manière uniforme. Un algorithme utilisant ce groupe accède aux lois de groupe par le biais d'oracles : il ne peut pas prévoir les réponses des oracles.

Des algorithmes génériques aux groupes génériques

Un algorithme générique (Shoup, Eurocrypt'1997) est une succession de requêtes (addition ou opposé) dans un groupe. Dans un sens plus large, on peut vouloir autoriser des tests, des boucles, mais rien ne doit permettre à l'algorithme de "prédire" des réponses à certaines requêtes.

La probabilité de succès d'un algorithme générique, sur un problème donné, est déterminée par la proportion de cas favorables dans l'ensemble des groupes compatibles avec les réponses obtenues par l'algorithme.

Le groupe générique est un groupe dont la loi est tirée aléatoirement de manière uniforme. Un algorithme utilisant ce groupe accède aux lois de groupe par le biais d'oracles : il ne peut pas prévoir les réponses des oracles.

Des algorithmes génériques aux groupes génériques

Un algorithme générique (Shoup, Eurocrypt'1997) est une succession de requêtes (addition ou opposé) dans un groupe. Dans un sens plus large, on peut vouloir autoriser des tests, des boucles, mais rien ne doit permettre à l'algorithme de "prédire" des réponses à certaines requêtes.

La probabilité de succès d'un algorithme générique, sur un problème donné, est déterminée par la proportion de cas favorables dans l'ensemble des groupes compatibles avec les réponses obtenues par l'algorithme.

Le groupe générique est un groupe dont la loi est tirée aléatoirement de manière uniforme. Un algorithme utilisant ce groupe accède aux lois de groupe par le biais d'oracles : il ne peut pas prévoir les réponses des oracles.

Exemples (1)

L'algorithme générique "pas de bébé, pas de géant"

BSGS(n,z,g,x)

$m = \lceil \sqrt{n} \rceil$
$t_1 = z$
for $i_1 = 0 \ldots (m-1)$
    insert (t_1, i_1) in hash_table
    $t_1 = t_1 + g$
$t_1 = -t_1$
$t_2 = x$
for $i_2 = 0 \ldots m$
    $i_1 = \text{search}(t_2, \text{hash\_table})$
    if $i_1 \neq$ not_found
        return $(i_1 + m.i_2)$
    $t_2 = t_2 + t_1$

On insère progressivement $(i_1.g, i_1)$ dans hash_table où $i_1 \in \{1, \ldots, m-1\}$

On cherche progressivement $(x - i_2.m.g)$ dans hash_table où $i_2 \in \{1, \ldots, m\}$

Exemples (1)

L'algorithme générique "pas de bébé, pas de géant"

BSGS(n,z,g,x)

$m = \lceil \sqrt{n} \rceil$
$t_1 = z$
for $i_1 = 0 \ldots (m-1)$
    insert (t_1, i_1) in hash_table
    $t_1 = t_1 + g$
$t_1 = -t_1$
$t_2 = x$
for $i_2 = 0 \ldots m$
    $i_1 = \text{search}(t_2, \text{hash\_table})$
    if $i_1 \neq$ not_found
        return $(i_1 + m.i_2)$
    $t_2 = t_2 + t_1$

On insère progressivement $(i_1.g, i_1)$ dans hash_table où $i_1 \in \{1, \ldots, m-1\}$

On cherche progressivement $(x - i_2.m.g)$ dans hash_table où $i_2 \in \{1, \ldots, m\}$

Exemples (2)

L'algorithme non-générique basé sur Euclide étendu

Euclide(a,b)

if $a = 1$
  return $(1, 0)$
$(\lambda, \mu) = \text{division}(b, a)$
$(u, v) = \text{Euclide}(\mu, a)$
return $(v - \lambda u, u)$

$b = \lambda.a + \mu$, avec $0 \leq \mu < a$
$u.\mu + v.a = 1$
donc $(v - \lambda.u).a + u.b = 1$

EuclideLog(n, z, g, x)

$(u, v) = \text{Euclide}(g, n)$
return $(u.x)$

$u = g^{-1}$ dans $\mathbb{Z}/n\mathbb{Z}$
donc $(u.x).g = x$ dans $\mathbb{Z}/n\mathbb{Z}$

Exemples (2)

L'algorithme non-générique basé sur Euclide étendu

Euclide(a,b)

if $a = 1$
  return $(1, 0)$
$(\lambda, \mu) = \text{division}(b, a)$
$(u, v) = \text{Euclide}(\mu, a)$
return $(v - \lambda u, u)$

$b = \lambda.a + \mu$, avec $0 \leq \mu < a$
$u.\mu + v.a = 1$
donc $(v - \lambda.u).a + u.b = 1$

EuclideLog(n, z, g, x)

$(u, v) = \text{Euclide}(g, n)$
return $(u.x)$

$u = g^{-1}$ dans $\mathbb{Z}/n\mathbb{Z}$
donc $(u.x).g = x$ dans $\mathbb{Z}/n\mathbb{Z}$

Plan de l'exposé

Famille de représentations de groupes cycliques

La donnée d'un groupe cyclique consiste usuellement en :

- un ensemble fini A, et son cardinal n,
- deux éléments g_A et 0_A de cet ensemble,
- une loi binaire $(+)$, et une loi unaire $(-)$.

Définition (Famille de représentations de groupes cycliques)

Une famille de représentations de groupes cycliques sur un langage L est :

- *un ensemble dénombrable de paramètres Ω,*
- *une fonction facilement calculable $c : \Omega \to \mathbb{N}^*$ non-bornée,*
- *un sous-ensemble L_α de L, de taille $c(\alpha)$, pour tout $\alpha \in \Omega$,*
- *un couple $(0_\alpha, g_\alpha) \in {L_\alpha}^2$, pour tout $\alpha \in \Omega$,*
- *deux algorithmes $+_\alpha$ et $-_\alpha$ sur L_α, pour tout $\alpha \in \Omega$.*

Pour tout $\alpha \in \Omega$, L_α muni des lois binaire $+_\alpha$ et unaire $-_\alpha$ est un groupe cyclique d'élément neutre 0_α et de générateur g_α.

Famille de représentations de groupes cycliques

La donnée d'un groupe cyclique consiste usuellement en :

- un ensemble fini A, et son cardinal n,
- deux éléments g_A et 0_A de cet ensemble,
- une loi binaire $(+)$, et une loi unaire $(-)$.

Définition (Famille de représentations de groupes cycliques)

Une famille de représentations de groupes cycliques sur un langage L est :

- *un ensemble dénombrable de paramètres Ω,*
- *une fonction facilement calculable $c : \Omega \to \mathbb{N}^*$ non-bornée,*
- *un sous-ensemble L_α de L, de taille $c(\alpha)$, pour tout $\alpha \in \Omega$,*
- *un couple $(0_\alpha, g_\alpha) \in {L_\alpha}^2$, pour tout $\alpha \in \Omega$,*
- *deux algorithmes $+_\alpha$ et $-_\alpha$ sur L_α, pour tout $\alpha \in \Omega$.*

Pour tout $\alpha \in \Omega$, L_α muni des lois binaire $+_\alpha$ et unaire $-_\alpha$ est un groupe cyclique d'élément neutre 0_α et de générateur g_α.

Famille de représentations de groupes cycliques

La donnée d'un groupe cyclique consiste usuellement en :

- un ensemble fini A, et son cardinal n,
- deux éléments g_A et 0_A de cet ensemble,
- une loi binaire $(+)$, et une loi unaire $(-)$.

Définition (Famille de représentations de groupes cycliques)

Une famille de représentations de groupes cycliques sur un langage L est :

- *un ensemble dénombrable de paramètres Ω,*
- *une fonction facilement calculable $c : \Omega \longrightarrow \mathbb{N}^*$ non-bornée,*
- *un sous-ensemble L_α de L, de taille $c(\alpha)$, pour tout $\alpha \in \Omega$,*
- *un couple $(0_\alpha, g_\alpha) \in {L_\alpha}^2$, pour tout $\alpha \in \Omega$,*
- *deux algorithmes $+_\alpha$ et $-_\alpha$ sur L_α, pour tout $\alpha \in \Omega$.*

Pour tout $\alpha \in \Omega$, L_α muni des lois binaire $+_\alpha$ et unaire $-_\alpha$ est un groupe cyclique d'élément neutre 0_α et de générateur g_α.

Famille de représentations de groupes cycliques

La donnée d'un groupe cyclique consiste usuellement en :

- un ensemble fini A, et son cardinal n,
- deux éléments g_A et 0_A de cet ensemble,
- une loi binaire $(+)$, et une loi unaire $(-)$.

Définition (Famille de représentations de groupes cycliques)

Une famille de représentations de groupes cycliques sur un langage L est :

- *un ensemble dénombrable de paramètres Ω,*
- *une fonction facilement calculable $c : \Omega \to \mathbb{N}^*$ non-bornée,*
- *un sous-ensemble L_α de L, de taille $c(\alpha)$, pour tout $\alpha \in \Omega$,*
- *un couple $(0_\alpha, g_\alpha) \in {L_\alpha}^2$, pour tout $\alpha \in \Omega$,*
- *deux algorithmes $+_\alpha$ et $-_\alpha$ sur L_α, pour tout $\alpha \in \Omega$.*

Pour tout $\alpha \in \Omega$, L_α muni des lois binaire $+_\alpha$ et unaire $-_\alpha$ est un groupe cyclique d'élément neutre 0_α et de générateur g_α.

Famille de représentations de groupes cycliques

La donnée d'un groupe cyclique consiste usuellement en :

- un ensemble fini A, et son cardinal n,
- deux éléments g_A et 0_A de cet ensemble,
- une loi binaire $(+)$, et une loi unaire $(-)$.

Définition (Famille de représentations de groupes cycliques)

Une famille de représentations de groupes cycliques sur un langage L est :

- *un ensemble dénombrable de paramètres Ω,*
- *une fonction facilement calculable $c : \Omega \longrightarrow \mathbb{N}^*$ non-bornée,*
- *un sous-ensemble L_α de L, de taille $c(\alpha)$, pour tout $\alpha \in \Omega$,*
- *un couple $(0_\alpha, g_\alpha) \in {L_\alpha}^2$, pour tout $\alpha \in \Omega$,*
- *deux algorithmes $+_\alpha$ et $-_\alpha$ sur L_α, pour tout $\alpha \in \Omega$.*

Pour tout $\alpha \in \Omega$, L_α muni des lois binaire $+_\alpha$ et unaire $-_\alpha$ est un groupe cyclique d'élément neutre 0_α et de générateur g_α.

Famille de représentations de groupes cycliques

La donnée d'un groupe cyclique consiste usuellement en :

- un ensemble fini A, et son cardinal n,
- deux éléments g_A et 0_A de cet ensemble,
- une loi binaire $(+)$, et une loi unaire $(-)$.

Définition (Famille de représentations de groupes cycliques)

Une famille de représentations de groupes cycliques sur un langage L est :

- *un ensemble dénombrable de paramètres Ω,*
- *une fonction facilement calculable $c : \Omega \longrightarrow \mathbb{N}^*$ non-bornée,*
- *un sous-ensemble L_α de L, de taille $c(\alpha)$, pour tout $\alpha \in \Omega$,*
- *un couple $(0_\alpha, g_\alpha) \in {L_\alpha}^2$, pour tout $\alpha \in \Omega$,*
- *deux algorithmes $+_\alpha$ et $-_\alpha$ sur L_α, pour tout $\alpha \in \Omega$.*

Pour tout $\alpha \in \Omega$, L_α muni des lois binaire $+_\alpha$ et unaire $-_\alpha$ est un groupe cyclique d'élément neutre 0_α et de générateur g_α.

Famille de représentations de groupes cycliques

La donnée d'un groupe cyclique consiste usuellement en :

- un ensemble fini A, et son cardinal n,
- deux éléments g_A et 0_A de cet ensemble,
- une loi binaire $(+)$, et une loi unaire $(-)$.

Définition (Famille de représentations de groupes cycliques)

Une famille de représentations de groupes cycliques sur un langage L est :

- *un ensemble dénombrable de paramètres Ω,*
- *une fonction facilement calculable $c : \Omega \longrightarrow \mathbb{N}^*$ non-bornée,*
- *un sous-ensemble L_α de L, de taille $c(\alpha)$, pour tout $\alpha \in \Omega$,*
- *un couple $(0_\alpha, g_\alpha) \in {L_\alpha}^2$, pour tout $\alpha \in \Omega$,*
- *deux algorithmes $+_\alpha$ et $-_\alpha$ sur L_α, pour tout $\alpha \in \Omega$.*

Pour tout $\alpha \in \Omega$, L_α muni des lois binaire $+_\alpha$ et unaire $-_\alpha$ est un groupe cyclique d'élément neutre 0_α et de générateur g_α.

Famille de représentations de groupes cycliques

La donnée d'un groupe cyclique consiste usuellement en :

- un ensemble fini A, et son cardinal n,
- deux éléments g_A et 0_A de cet ensemble,
- une loi binaire $(+)$, et une loi unaire $(-)$.

Définition (Famille de représentations de groupes cycliques)

Une famille de représentations de groupes cycliques sur un langage L est :

- *un ensemble dénombrable de paramètres Ω,*
- *une fonction facilement calculable $c : \Omega \longrightarrow \mathbb{N}^*$ non-bornée,*
- *un sous-ensemble L_α de L, de taille $c(\alpha)$, pour tout $\alpha \in \Omega$,*
- *un couple $(0_\alpha, g_\alpha) \in {L_\alpha}^2$, pour tout $\alpha \in \Omega$,*
- *deux algorithmes $+_\alpha$ et $-_\alpha$ sur L_α, pour tout $\alpha \in \Omega$.*

Pour tout $\alpha \in \Omega$, L_α muni des lois binaire $+_\alpha$ et unaire $-_\alpha$ est un groupe cyclique d'élément neutre 0_α et de générateur g_α.

Famille générique de groupes cycliques (1)

La famille générique de groupes cycliques d'ordre n repose sur un ensemble fini A, de cardinal n. Les éléments neutre et générateur, ainsi que les lois binaire et unaire sont construites à partir d'une bijection $f : \mathbb{Z}/n\mathbb{Z} \to A$, choisie aléatoirement et uniformément :

- $0_f = f(0)$ et $g_f = f(1)$,
- $\forall x \in A, -_f x = f[-f^{-1}(x)]$,
- $\forall (x, y) \in A^2, x +_f y = f[f^{-1}(x) + f^{-1}(y)]$.

Les lois ne sont données que sous la forme d'oracles, et ainsi, toutes les lois de groupe sont a priori possible.

Famille générique de groupes cycliques (1)

La famille générique de groupes cycliques d'ordre n repose sur un ensemble fini A, de cardinal n. Les éléments neutre et générateur, ainsi que les lois binaire et unaire sont construites à partir d'une bijection $f : \mathbb{Z}/n\mathbb{Z} \to A$, choisie aléatoirement et uniformément :

- $0_f = f(0)$ et $g_f = f(1)$,
- $\forall x \in A, -_f x = f[-f^{-1}(x)]$,
- $\forall (x, y) \in A^2, x +_f y = f[f^{-1}(x) + f^{-1}(y)]$.

Les lois ne sont données que sous la forme d'oracles, et ainsi, toutes les lois de groupe sont a priori possible.

Famille générique de groupes cycliques (1)

La famille générique de groupes cycliques d'ordre n repose sur un ensemble fini A, de cardinal n. Les éléments neutre et générateur, ainsi que les lois binaire et unaire sont construites à partir d'une bijection $f : \mathbb{Z}/n\mathbb{Z} \to A$, choisie aléatoirement et uniformément :

- $0_f = f(0)$ et $g_f = f(1)$,
- $\forall x \in A, -_f x = f[-f^{-1}(x)]$,
- $\forall (x, y) \in A^2, x +_f y = f[f^{-1}(x) + f^{-1}(y)]$.

Les lois ne sont données que sous la forme d'oracles, et ainsi, toutes les lois de groupe sont a priori possible.

Famille générique de groupes cycliques (1)

La famille générique de groupes cycliques d'ordre n repose sur un ensemble fini A, de cardinal n. Les éléments neutre et générateur, ainsi que les lois binaire et unaire sont construites à partir d'une bijection $f : \mathbb{Z}/n\mathbb{Z} \to A$, choisie aléatoirement et uniformément :

- $0_f = f(0)$ et $g_f = f(1)$,
- $\forall x \in A, -_f x = f[-f^{-1}(x)]$,
- $\forall (x, y) \in A^2, x +_f y = f[f^{-1}(x) + f^{-1}(y)]$.

Les lois ne sont données que sous la forme d'oracles, et ainsi, toutes les lois de groupe sont a priori possible.

Famille générique de groupes cycliques (1)

La famille générique de groupes cycliques d'ordre n repose sur un ensemble fini A, de cardinal n. Les éléments neutre et générateur, ainsi que les lois binaire et unaire sont construites à partir d'une bijection $f : \mathbb{Z}/n\mathbb{Z} \to A$, choisie aléatoirement et uniformément :

- $0_f = f(0)$ et $g_f = f(1)$,
- $\forall x \in A, -_f x = f[-f^{-1}(x)]$,
- $\forall (x, y) \in A^2, x +_f y = f[f^{-1}(x) + f^{-1}(y)]$.

Les lois ne sont données que sous la forme d'oracles, et ainsi, toutes les lois de groupe sont a priori possible.

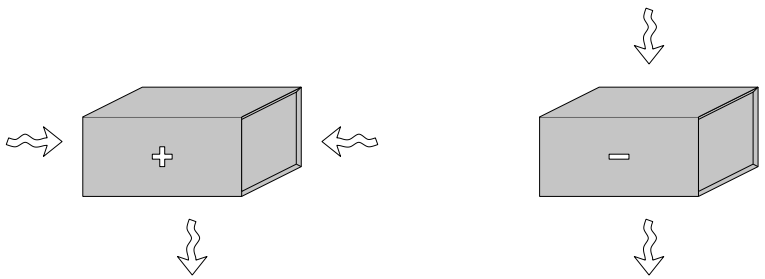
Famille générique de groupes cycliques (2)

Définition (Famille générique de groupes cycliques)

La famille générique de groupes cycliques peut être vue comme une famille de représentations de groupes cycliques sur $\{0,1\}^*$ *:*

- $\Omega = \{(n, 0_f, g_f), n \in \mathbb{N}^*, (0_f, g_f) \in B(n)^2\}$,
- $c : (n, 0_f, g_f) \in \Omega \mapsto n \in \mathbb{N}^*$,
- $\forall \alpha = (n, 0_f, g_f) \in \Omega, L_\alpha = B(n)$,
- $\forall \alpha = (n, 0_f, g_f) \in \Omega, (0_\alpha, g_\alpha) = (0_f, g_f)$.

Les lois $+_\alpha$ *et* $-_\alpha$*, construites à partir d'une bijection* $f : \mathbb{Z}/n\mathbb{Z} \to B(n)$ *vérifiant* $0_\alpha = f(0)$ *et* $g_\alpha = f(1)$*, sont données sous la forme d'oracles.*



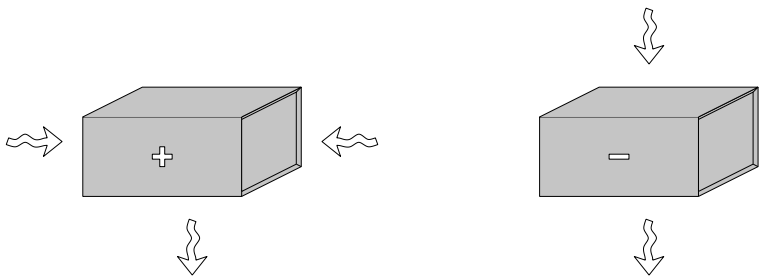
Famille générique de groupes cycliques (2)

Définition (Famille générique de groupes cycliques)

La famille générique de groupes cycliques peut être vue comme une famille de représentations de groupes cycliques sur $\{0,1\}^*$ *:*

- $\Omega = \{(n, 0_f, g_f), n \in \mathbb{N}^*, (0_f, g_f) \in B(n)^2\}$,
- $c : (n, 0_f, g_f) \in \Omega \mapsto n \in \mathbb{N}^*$,
- $\forall \alpha = (n, 0_f, g_f) \in \Omega, L_\alpha = B(n)$,
- $\forall \alpha = (n, 0_f, g_f) \in \Omega, (0_\alpha, g_\alpha) = (0_f, g_f)$.

Les lois $+_\alpha$ *et* $-_\alpha$*, construites à partir d'une bijection* $f : \mathbb{Z}/n\mathbb{Z} \to B(n)$ *vérifiant* $0_\alpha = f(0)$ *et* $g_\alpha = f(1)$*, sont données sous la forme d'oracles.*



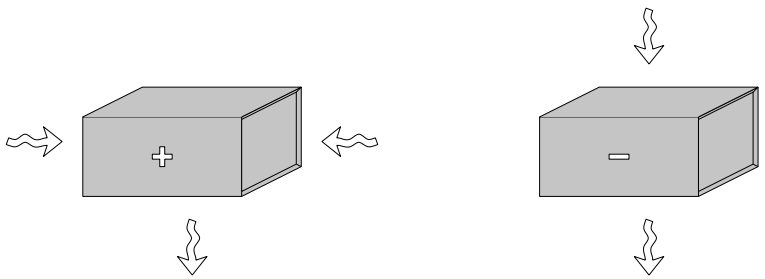
Famille générique de groupes cycliques (2)

Définition (Famille générique de groupes cycliques)

La famille générique de groupes cycliques peut être vue comme une famille de représentations de groupes cycliques sur $\{0,1\}^*$ *:*

- $\Omega = \{(n, 0_f, g_f), n \in \mathbb{N}^*, (0_f, g_f) \in B(n)^2\}$,
- $c : (n, 0_f, g_f) \in \Omega \mapsto n \in \mathbb{N}^*$,
- $\forall \alpha = (n, 0_f, g_f) \in \Omega, L_\alpha = B(n)$,
- $\forall \alpha = (n, 0_f, g_f) \in \Omega, (0_\alpha, g_\alpha) = (0_f, g_f)$.

Les lois $+_\alpha$ *et* $-_\alpha$*, construites à partir d'une bijection* $f : \mathbb{Z}/n\mathbb{Z} \to B(n)$ *vérifiant* $0_\alpha = f(0)$ *et* $g_\alpha = f(1)$*, sont données sous la forme d'oracles.*



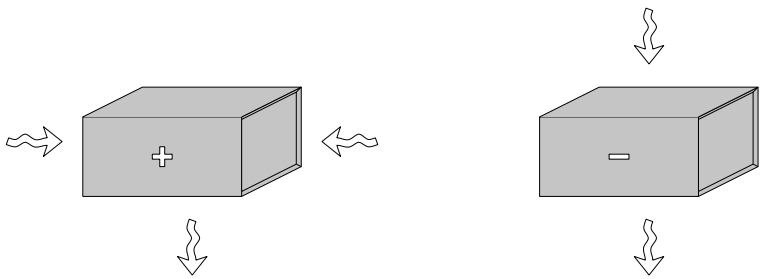
Famille générique de groupes cycliques (2)

Définition (Famille générique de groupes cycliques)

La famille générique de groupes cycliques peut être vue comme une famille de représentations de groupes cycliques sur $\{0,1\}^*$ *:*

- $\Omega = \{(n, 0_f, g_f), n \in \mathbb{N}^*, (0_f, g_f) \in B(n)^2\}$,
- $c : (n, 0_f, g_f) \in \Omega \mapsto n \in \mathbb{N}^*$,
- $\forall \alpha = (n, 0_f, g_f) \in \Omega, L_\alpha = B(n)$,
- $\forall \alpha = (n, 0_f, g_f) \in \Omega, (0_\alpha, g_\alpha) = (0_f, g_f)$.

Les lois $+_\alpha$ *et* $-_\alpha$*, construites à partir d'une bijection* $f : \mathbb{Z}/n\mathbb{Z} \to B(n)$ *vérifiant* $0_\alpha = f(0)$ *et* $g_\alpha = f(1)$*, sont données sous la forme d'oracles.*



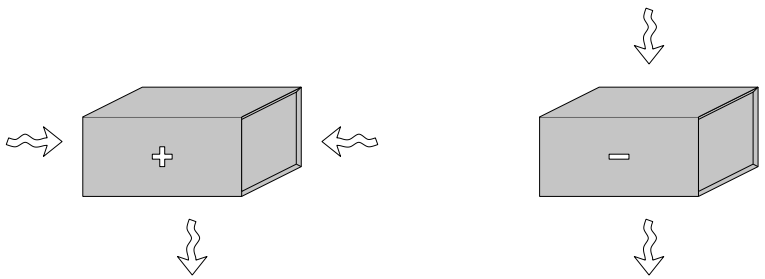
Famille générique de groupes cycliques (2)

Définition (Famille générique de groupes cycliques)

La famille générique de groupes cycliques peut être vue comme une famille de représentations de groupes cycliques sur $\{0,1\}^*$ *:*

- $\Omega = \{(n, 0_f, g_f), n \in \mathbb{N}^*, (0_f, g_f) \in B(n)^2\}$,
- $c : (n, 0_f, g_f) \in \Omega \mapsto n \in \mathbb{N}^*$,
- $\forall \alpha = (n, 0_f, g_f) \in \Omega, L_\alpha = B(n)$,
- $\forall \alpha = (n, 0_f, g_f) \in \Omega, (0_\alpha, g_\alpha) = (0_f, g_f)$.

Les lois $+_\alpha$ *et* $-_\alpha$*, construites à partir d'une bijection* $f : \mathbb{Z}/n\mathbb{Z} \to B(n)$ *vérifiant* $0_\alpha = f(0)$ *et* $g_\alpha = f(1)$*, sont données sous la forme d'oracles.*



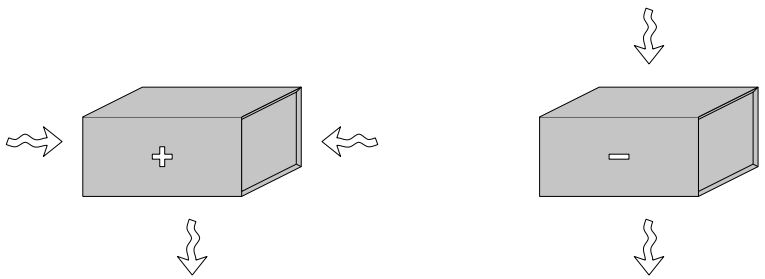
Famille générique de groupes cycliques (2)

Définition (Famille générique de groupes cycliques)

La famille générique de groupes cycliques peut être vue comme une famille de représentations de groupes cycliques sur $\{0,1\}^*$ *:*

- $\Omega = \{(n, 0_f, g_f), n \in \mathbb{N}^*, (0_f, g_f) \in B(n)^2\}$,
- $c : (n, 0_f, g_f) \in \Omega \mapsto n \in \mathbb{N}^*$,
- $\forall \alpha = (n, 0_f, g_f) \in \Omega, L_\alpha = B(n)$,
- $\forall \alpha = (n, 0_f, g_f) \in \Omega, (0_\alpha, g_\alpha) = (0_f, g_f)$.

Les lois $+_\alpha$ *et* $-_\alpha$*, construites à partir d'une bijection* $f : \mathbb{Z}/n\mathbb{Z} \to B(n)$ *vérifiant* $0_\alpha = f(0)$ *et* $g_\alpha = f(1)$*, sont données sous la forme d'oracles.*



Plan de l'exposé

Problèmes standards sur une famille de représentations

On considère une famille de représentations de groupes cycliques sur L.

Un algorithme résolvant le logarithme discret sur cette famille :

- reçoit en entrée un paramètre $\alpha \in \Omega$, un élément $x \in L_\alpha$,
- calcule $\log_{g_\alpha}(x)$ dans le groupe L_α muni de ses lois $+_\alpha$ et $-_\alpha$.

Un algorithme résolvant le problème Diffie-Hellman :

- reçoit en entrée un paramètre $\alpha \in \Omega$, un couple $(x, y) \in {L_\alpha}^2$,
- calcule $\log_{g_\alpha}(x).y$ dans le groupe L_α muni de ses lois $+_\alpha$ et $-_\alpha$.

Un algorithme résolvant le problème décisionnel Diffie-Hellman :

- reçoit en entrée un paramètre $\alpha \in \Omega$, un triplet $(x, y, z) \in {L_\alpha}^3$,
- décide si $z = \log_{g_\alpha}(x).y$ dans le groupe L_α muni de ses lois $+_\alpha$ et $-_\alpha$.

Problèmes standards sur une famille de représentations

On considère une famille de représentations de groupes cycliques sur L.

Un algorithme résolvant le logarithme discret sur cette famille :

- reçoit en entrée un paramètre $\alpha \in \Omega$, un élément $x \in L_\alpha$,
- calcule $\log_{g_\alpha}(x)$ dans le groupe L_α muni de ses lois $+_\alpha$ et $-_\alpha$.

Un algorithme résolvant le problème Diffie-Hellman :

- reçoit en entrée un paramètre $\alpha \in \Omega$, un couple $(x, y) \in {L_\alpha}^2$,
- calcule $\log_{g_\alpha}(x).y$ dans le groupe L_α muni de ses lois $+_\alpha$ et $-_\alpha$.

Un algorithme résolvant le problème décisionnel Diffie-Hellman :

- reçoit en entrée un paramètre $\alpha \in \Omega$, un triplet $(x, y, z) \in {L_\alpha}^3$,
- décide si $z = \log_{g_\alpha}(x).y$ dans le groupe L_α muni de ses lois $+_\alpha$ et $-_\alpha$.

Problèmes standards sur une famille de représentations

On considère une famille de représentations de groupes cycliques sur L.

Un algorithme résolvant le logarithme discret sur cette famille :

- reçoit en entrée un paramètre $\alpha \in \Omega$, un élément $x \in L_\alpha$,
- calcule $\log_{g_\alpha}(x)$ dans le groupe L_α muni de ses lois $+_\alpha$ et $-_\alpha$.

Un algorithme résolvant le problème Diffie-Hellman :

- reçoit en entrée un paramètre $\alpha \in \Omega$, un couple $(x, y) \in {L_\alpha}^2$,
- calcule $\log_{g_\alpha}(x).y$ dans le groupe L_α muni de ses lois $+_\alpha$ et $-_\alpha$.

Un algorithme résolvant le problème décisionnel Diffie-Hellman :

- reçoit en entrée un paramètre $\alpha \in \Omega$, un triplet $(x, y, z) \in {L_\alpha}^3$,
- décide si $z = \log_{g_\alpha}(x).y$ dans le groupe L_α muni de ses lois $+_\alpha$ et $-_\alpha$.

Problèmes standards sur la famille générique (1)

Dans le cadre de la famille générique de groupes cycliques, l'algorithme a accès à des oracles au lieu de pouvoir lui-même calculer les lois de groupe.

Un algorithme résolvant le logarithme discret sur la famille générique :

- reçoit en entrée une taille $n \in \mathbb{N}^*$, des éléments $(0_f, g_f, x) \in B(n)^3$,
- a accès à des oracles $+_f$ et $-_f$ construits à partir d'une bijection f,
- calcule $\log_{g_f}(x)$ dans le groupe $B(n)$ muni des lois $+_f$ et $-_f$.

Problèmes standards sur la famille générique (1)

Dans le cadre de la famille générique de groupes cycliques, l'algorithme a accès à des oracles au lieu de pouvoir lui-même calculer les lois de groupe.

Un algorithme résolvant le logarithme discret sur la famille générique :

- reçoit en entrée une taille $n \in \mathbb{N}^*$, des éléments $(0_f, g_f, x) \in B(n)^3$,
- a accès à des oracles $+_f$ et $-_f$ construits à partir d'une bijection f,
- calcule $\log_{g_f}(x)$ dans le groupe $B(n)$ muni des lois $+_f$ et $-_f$.

Problèmes standards sur la famille générique (2)

Un algorithme résolvant le problème Diffie-Hellman :

- reçoit en entrée $n \in \mathbb{N}^*$, $(0_f, g_f, x, y) \in B(n)^4$,
- a accès à des oracles $+_f$ et $-_f$ construits à partir d'une bijection f,
- calcule $\log_{g_f}(x).y$ dans le groupe $B(n)$ muni des lois $+_f$ et $-_f$.

Un algorithme résolvant le problème décisionnel Diffie-Hellman :

- reçoit en entrée $n \in \mathbb{N}^*$, $(0_f, g_f, x, y, z) \in B(n)^5$,
- a accès à des oracles $+_f$ et $-_f$ construits à partir d'une bijection f,
- décide si $z = \log_{g_f}(x).y$ dans le groupe $B(n)$ muni des lois $+_f$ et $-_f$.

Problèmes standards sur la famille générique (2)

Un algorithme résolvant le problème Diffie-Hellman :

- reçoit en entrée $n \in \mathbb{N}^*$, $(0_f, g_f, x, y) \in B(n)^4$,
- a accès à des oracles $+_f$ et $-_f$ construits à partir d'une bijection f,
- calcule $\log_{g_f}(x).y$ dans le groupe $B(n)$ muni des lois $+_f$ et $-_f$.

Un algorithme résolvant le problème décisionnel Diffie-Hellman :

- reçoit en entrée $n \in \mathbb{N}^*$, $(0_f, g_f, x, y, z) \in B(n)^5$,
- a accès à des oracles $+_f$ et $-_f$ construits à partir d'une bijection f,
- décide si $z = \log_{g_f}(x).y$ dans le groupe $B(n)$ muni des lois $+_f$ et $-_f$.

Plan de l'exposé

Recherche d'un couplage

Soient g, g' et g'' les générateurs de 3 groupes G, G' et G'' de même ordre n. Ces groupes sont structurellement des $(\mathbb{Z}/n\mathbb{Z})$-modules.

Il existe une unique application $\mathbb{Z}/n\mathbb{Z}$-bilinéaire $e : G \times G' \longrightarrow G''$ telle que : $e(g, g') = g''$.

Dans le cas $G = G' = B(n)$ muni de lois induites par $f : \mathbb{Z}/n\mathbb{Z} \longrightarrow B(n)$, et $G'' = B(n)$ muni de lois induites par $h : \mathbb{Z}/n\mathbb{Z} \longrightarrow B(n)$, l'application suivante convient :

$$e_{f,h} : (x, y) \in B(n)^2 \mapsto h[f^{-1}(x).f^{-1}(y)] \in B(n).$$

Recherche d'un couplage

Soient g, g' et g'' les générateurs de 3 groupes G, G' et G'' de même ordre n. Ces groupes sont structurellement des $(\mathbb{Z}/n\mathbb{Z})$-modules.

Il existe une unique application $\mathbb{Z}/n\mathbb{Z}$-bilinéaire $e : G \times G' \to G''$ telle que : $e(g, g') = g''$.

Dans le cas $G = G' = B(n)$ muni de lois induites par $f : \mathbb{Z}/n\mathbb{Z} \to B(n)$, et $G'' = B(n)$ muni de lois induites par $h : \mathbb{Z}/n\mathbb{Z} \to B(n)$, l'application suivante convient :

$$e_{f,h} : (x, y) \in B(n)^2 \mapsto h[f^{-1}(x).f^{-1}(y)] \in B(n).$$

Recherche d'un couplage

Soient g, g' et g'' les générateurs de 3 groupes G, G' et G'' de même ordre n. Ces groupes sont structurellement des $(\mathbb{Z}/n\mathbb{Z})$-modules.

Il existe une unique application $\mathbb{Z}/n\mathbb{Z}$-bilinéaire $e : G \times G' \to G''$ telle que : $e(g, g') = g''$.

Dans le cas $G = G' = B(n)$ muni de lois induites par $f : \mathbb{Z}/n\mathbb{Z} \to B(n)$, et $G'' = B(n)$ muni de lois induites par $h : \mathbb{Z}/n\mathbb{Z} \to B(n)$, l'application suivante convient :

$$e_{f,h} : (x, y) \in B(n)^2 \mapsto h[f^{-1}(x).f^{-1}(y)] \in B(n).$$

Plan de l'exposé

Famille de représentations de groupes avec couplage

Définition (Famille de représentations de groupes avec couplage)

Une famille de représentations de groupes cycliques avec couplage sur deux langages L et M est la donnée de :

- *deux familles de représentations de groupes cycliques :*
  - $(\Gamma, \{(L_\gamma, +_\gamma, -_\gamma, 0_\gamma, g_\gamma), \gamma \in \Gamma\})$ sur le langage L,
  - $(\Delta, \{(M_\delta, +_\delta, -_\delta, 0_\delta, g_\delta), \delta \in \Delta\})$ sur le langage M,
- *un espace de paramètres $\Omega \subset \Gamma \times \Delta$,*
- *un couplage facilement calculable, $e_\alpha : L_\gamma \times L_\gamma \to M_\delta$, tel que $e(g_\gamma, g_\gamma) = g_\delta$, pour tout $\alpha = (\gamma, \delta) \in \Omega$.*

Famille de représentations de groupes avec couplage

Définition (Famille de représentations de groupes avec couplage)

Une famille de représentations de groupes cycliques avec couplage sur deux langages L et M est la donnée de :

- *deux familles de représentations de groupes cycliques :*
  - $(\Gamma, \{(L_\gamma, +_\gamma, -_\gamma, 0_\gamma, g_\gamma), \gamma \in \Gamma\})$ *sur le langage L,*
  - $(\Delta, \{(M_\delta, +_\delta, -_\delta, 0_\delta, g_\delta), \delta \in \Delta\})$ *sur le langage M,*
- *un espace de paramètres $\Omega \subset \Gamma \times \Delta$,*
- *un couplage facilement calculable, $e_\alpha : L_\gamma \times L_\gamma \to M_\delta$, tel que $e(g_\gamma, g_\gamma) = g_\delta$, pour tout $\alpha = (\gamma, \delta) \in \Omega$.*

Famille de représentations de groupes avec couplage

Définition (Famille de représentations de groupes avec couplage)

Une famille de représentations de groupes cycliques avec couplage sur deux langages L et M est la donnée de :

- *deux familles de représentations de groupes cycliques :*
  - $(\Gamma, \{(L_\gamma, +_\gamma, -_\gamma, 0_\gamma, g_\gamma), \gamma \in \Gamma\})$ *sur le langage L,*
  - $(\Delta, \{(M_\delta, +_\delta, -_\delta, 0_\delta, g_\delta), \delta \in \Delta\})$ *sur le langage M,*
- *un espace de paramètres $\Omega \subset \Gamma \times \Delta$,*
- *un couplage facilement calculable, $e_\alpha : L_\gamma \times L_\gamma \to M_\delta$, tel que $e(g_\gamma, g_\gamma) = g_\delta$, pour tout $\alpha = (\gamma, \delta) \in \Omega$.*

Famille de représentations de groupes avec couplage

Définition (Famille de représentations de groupes avec couplage)

Une famille de représentations de groupes cycliques avec couplage sur deux langages L et M est la donnée de :

- *deux familles de représentations de groupes cycliques :*
  - $(\Gamma, \{(L_\gamma, +_\gamma, -_\gamma, 0_\gamma, g_\gamma), \gamma \in \Gamma\})$ *sur le langage L,*
  - $(\Delta, \{(M_\delta, +_\delta, -_\delta, 0_\delta, g_\delta), \delta \in \Delta\})$ *sur le langage M,*
- *un espace de paramètres $\Omega \subset \Gamma \times \Delta$,*
- *un couplage facilement calculable, $e_\alpha : L_\gamma \times L_\gamma \to M_\delta$, tel que $e(g_\gamma, g_\gamma) = g_\delta$, pour tout $\alpha = (\gamma, \delta) \in \Omega$.*

Famille générique de groupes cycliques avec couplage (1)

Définition (Famille générique de groupes cycliques avec couplage)

La famille générique de groupes cycliques avec couplage peut être vue comme une famille de représentations de groupes cycliques avec couplage sur les langages $\{0,1\}^*$ *et* $\{0,1\}^*$ *:*

- *on utilise deux fois la famille générique de groupes cycliques,*
- $\Omega = \{((n, 0_f, g_f), (n, 0_h, g_h)), n \in \mathbb{N}^*, (0_f, g_f, 0_h, g_h) \in B(n)^4\}$,
- *les lois* $(+_f, -_f, +_h, -_h, e_{f,h})$ *sont toutes fournies sous la forme d'oracles, contruits à partir de :*

$$\begin{cases} f : \mathbb{Z}/n\mathbb{Z} \to B(n), \text{ bijection telle que } 0_f = f(0), g_f = f(1), \\ h : \mathbb{Z}/n\mathbb{Z} \to B(n), \text{ bijection telle que } 0_h = h(0), g_h = h(1). \end{cases}$$

Famille générique de groupes cycliques avec couplage (1)

Définition (Famille générique de groupes cycliques avec couplage)

La famille générique de groupes cycliques avec couplage peut être vue comme une famille de représentations de groupes cycliques avec couplage sur les langages $\{0,1\}^*$ *et* $\{0,1\}^*$ *:*

- *on utilise deux fois la famille générique de groupes cycliques,*
- $\Omega = \{((n, 0_f, g_f), (n, 0_h, g_h)), n \in \mathbb{N}^*, (0_f, g_f, 0_h, g_h) \in B(n)^4\}$,
- *les lois* $(+_f, -_f, +_h, -_h, e_{f,h})$ *sont toutes fournies sous la forme d'oracles, contruits à partir de :*

$$\begin{cases} f : \mathbb{Z}/n\mathbb{Z} \to B(n), \text{ bijection telle que } 0_f = f(0), g_f = f(1), \\ h : \mathbb{Z}/n\mathbb{Z} \to B(n), \text{ bijection telle que } 0_h = h(0), g_h = h(1). \end{cases}$$

Famille générique de groupes cycliques avec couplage (1)

Définition (Famille générique de groupes cycliques avec couplage)

La famille générique de groupes cycliques avec couplage peut être vue comme une famille de représentations de groupes cycliques avec couplage sur les langages $\{0,1\}^*$ *et* $\{0,1\}^*$ *:*

- *on utilise deux fois la famille générique de groupes cycliques,*
- $\Omega = \{((n, 0_f, g_f), (n, 0_h, g_h)), n \in \mathbb{N}^*, (0_f, g_f, 0_h, g_h) \in B(n)^4\}$,
- *les lois* $(+_f, -_f, +_h, -_h, e_{f,h})$ *sont toutes fournies sous la forme d'oracles, contruits à partir de :*

$$\begin{cases} f : \mathbb{Z}/n\mathbb{Z} \to B(n), \text{ bijection telle que } 0_f = f(0), g_f = f(1), \\ h : \mathbb{Z}/n\mathbb{Z} \to B(n), \text{ bijection telle que } 0_h = h(0), g_h = h(1). \end{cases}$$

Famille générique de groupes cycliques avec couplage (1)

Définition (Famille générique de groupes cycliques avec couplage)

La famille générique de groupes cycliques avec couplage peut être vue comme une famille de représentations de groupes cycliques avec couplage sur les langages $\{0,1\}^*$ *et* $\{0,1\}^*$ *:*

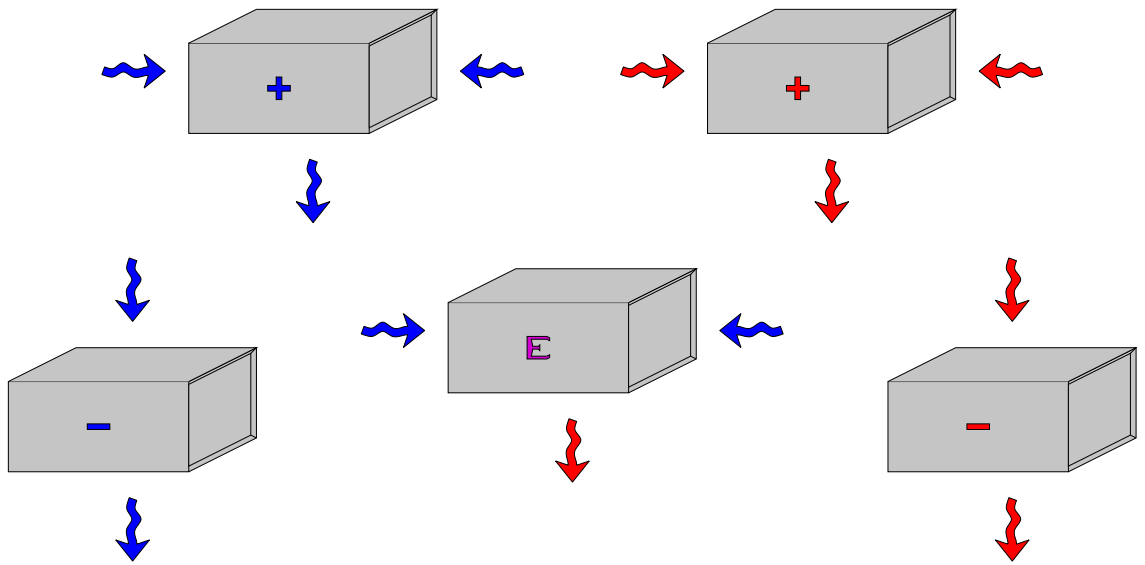
- *on utilise deux fois la famille générique de groupes cycliques,*
- $\Omega = \{((n, 0_f, g_f), (n, 0_h, g_h)), n \in \mathbb{N}^*, (0_f, g_f, 0_h, g_h) \in B(n)^4\}$,
- *les lois* $(+_f, -_f, +_h, -_h, e_{f,h})$ *sont toutes fournies sous la forme d'oracles, contruits à partir de :*

$$\begin{cases} f : \mathbb{Z}/n\mathbb{Z} \to B(n), \text{ bijection telle que } 0_f = f(0), g_f = f(1), \\ h : \mathbb{Z}/n\mathbb{Z} \to B(n), \text{ bijection telle que } 0_h = h(0), g_h = h(1). \end{cases}$$

Famille générique de groupes cycliques avec couplage (2)

Les oracles :

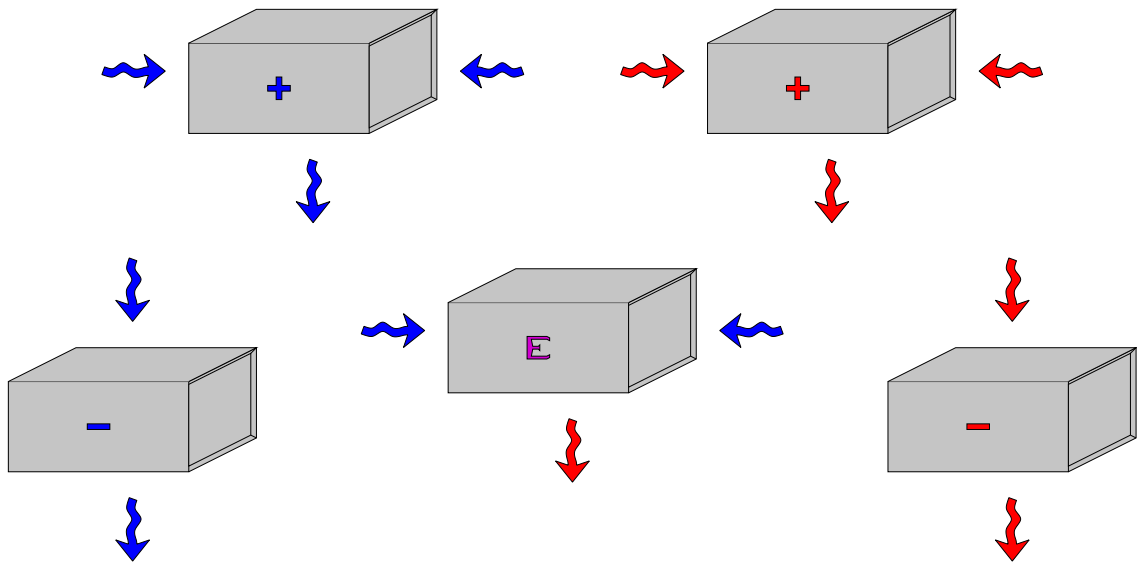
- $\forall (x,y) \in B(n)^2, x +_f y = f[f^{-1}(x) + f^{-1}(y)]$,
- $\forall x \in B(n), -_f x = f[-f^{-1}(x)]$,
- $\forall (x,y) \in B(n)^2, x +_h y = h[h^{-1}(x) + h^{-1}(y)]$,
- $\forall x \in B(n), -_h x = h[-h^{-1}(x)]$,
- $\forall (x,y) \in B(n)^2, e_{f,h}(x,y) = h[f^{-1}(x).f^{-1}(y)]$,



Famille générique de groupes cycliques avec couplage (2)

Les oracles :

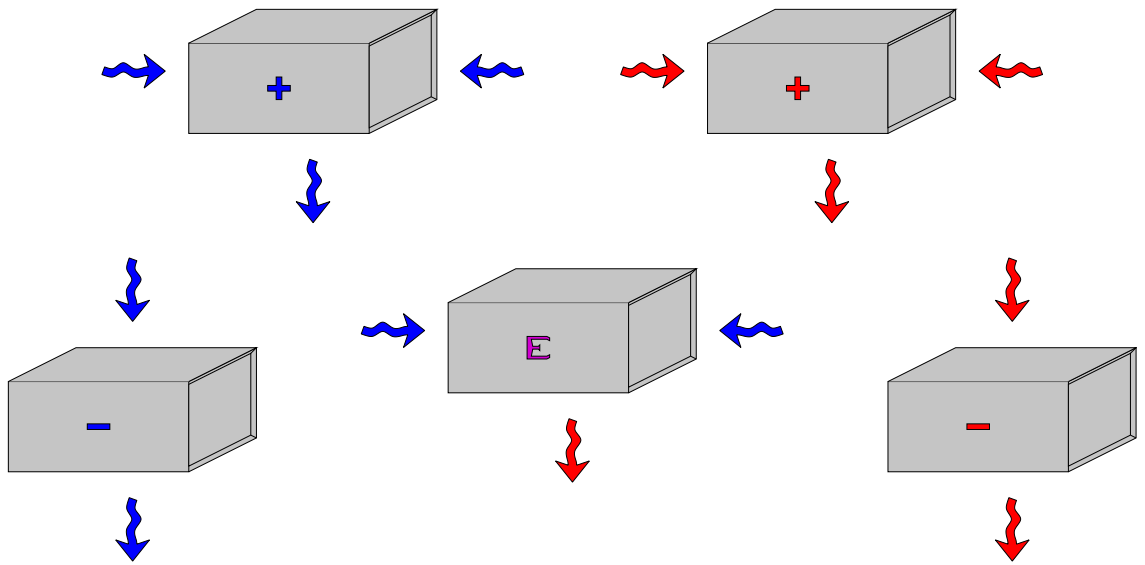
- $\forall (x,y) \in B(n)^2, x +_f y = f[f^{-1}(x) + f^{-1}(y)]$,
- $\forall x \in B(n), -_f x = f[-f^{-1}(x)]$,
- $\forall (x,y) \in B(n)^2, x +_h y = h[h^{-1}(x) + h^{-1}(y)]$,
- $\forall x \in B(n), -_h x = h[-h^{-1}(x)]$,
- $\forall (x,y) \in B(n)^2, e_{f,h}(x,y) = h[f^{-1}(x).f^{-1}(y)]$,



Famille générique de groupes cycliques avec couplage (2)

Les oracles :

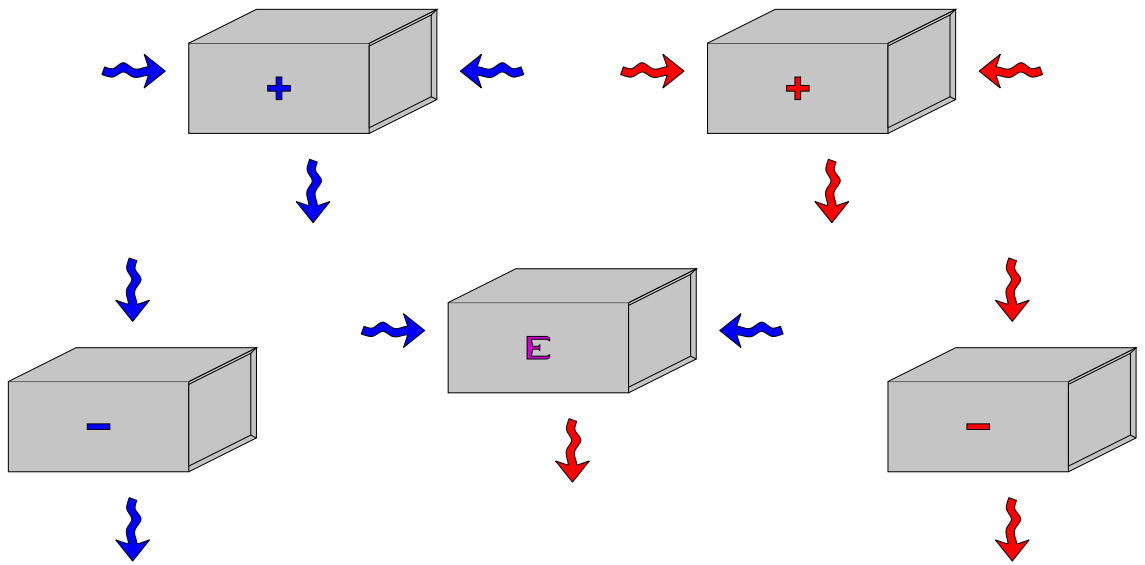
- $\forall (x,y) \in B(n)^2, x +_f y = f[f^{-1}(x) + f^{-1}(y)]$,
- $\forall x \in B(n), -_f x = f[-f^{-1}(x)]$,
- $\forall (x,y) \in B(n)^2, x +_h y = h[h^{-1}(x) + h^{-1}(y)]$,
- $\forall x \in B(n), -_h x = h[-h^{-1}(x)]$,
- $\forall (x,y) \in B(n)^2, e_{f,h}(x,y) = h[f^{-1}(x).f^{-1}(y)]$,



Famille générique de groupes cycliques avec couplage (2)

Les oracles :

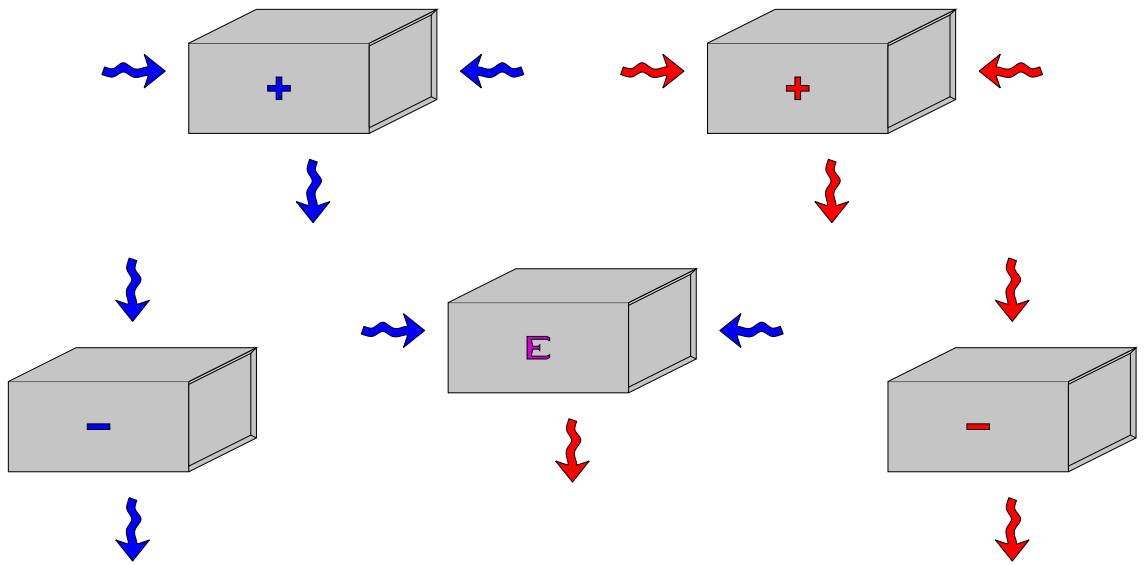
- $\forall (x,y) \in B(n)^2, x +_f y = f[f^{-1}(x) + f^{-1}(y)]$,
- $\forall x \in B(n), -_f x = f[-f^{-1}(x)]$,
- $\forall (x,y) \in B(n)^2, x +_h y = h[h^{-1}(x) + h^{-1}(y)]$,
- $\forall x \in B(n), -_h x = h[-h^{-1}(x)]$,
- $\forall (x,y) \in B(n)^2, e_{f,h}(x,y) = h[f^{-1}(x).f^{-1}(y)]$,



Famille générique de groupes cycliques avec couplage (2)

Les oracles :

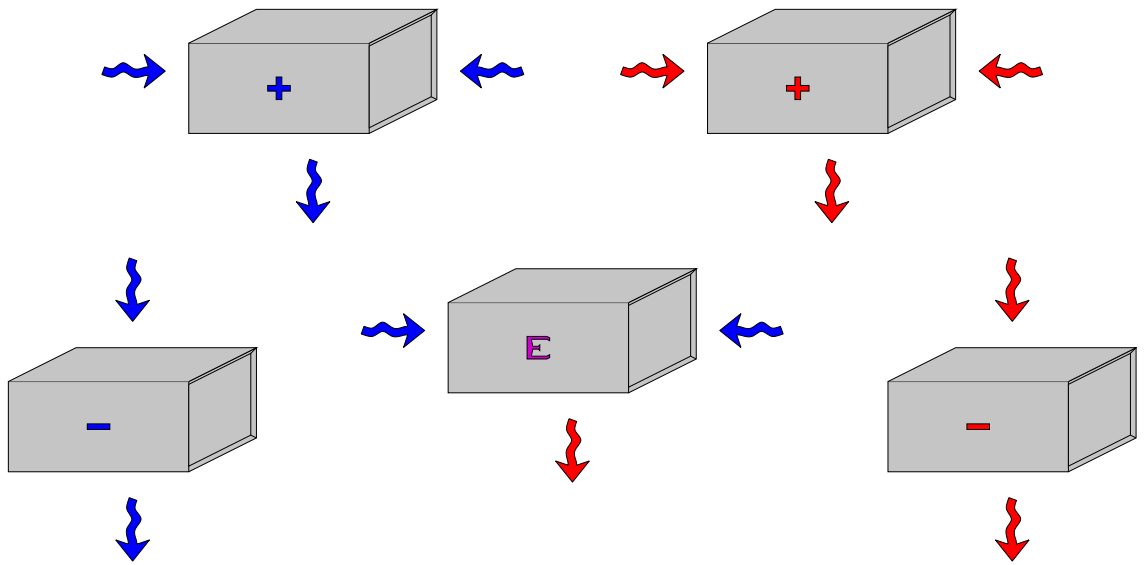
- $\forall (x,y) \in B(n)^2, x +_f y = f[f^{-1}(x) + f^{-1}(y)]$,
- $\forall x \in B(n), -_f x = f[-f^{-1}(x)]$,
- $\forall (x,y) \in B(n)^2, x +_h y = h[h^{-1}(x) + h^{-1}(y)]$,
- $\forall x \in B(n), -_h x = h[-h^{-1}(x)]$,
- $\forall (x,y) \in B(n)^2, e_{f,h}(x,y) = h[f^{-1}(x).f^{-1}(y)]$,



Famille générique de groupes cycliques avec couplage (2)

Les oracles :

- $\forall (x,y) \in B(n)^2, x +_f y = f[f^{-1}(x) + f^{-1}(y)]$,
- $\forall x \in B(n), -_f x = f[-f^{-1}(x)]$,
- $\forall (x,y) \in B(n)^2, x +_h y = h[h^{-1}(x) + h^{-1}(y)]$,
- $\forall x \in B(n), -_h x = h[-h^{-1}(x)]$,
- $\forall (x,y) \in B(n)^2, e_{f,h}(x,y) = h[f^{-1}(x).f^{-1}(y)]$,



Plan de l'exposé

Problèmes standards sur une famille avec couplage

On considère une famille de représentations de groupes cycliques avec couplages sur L et M.

Un algorithme résolvant le problème Diffie-Hellman bilinéaire :

- reçoit en entrée un paramètre $\alpha \in \Omega$, un triplet $(w, x, y) \in {L_\gamma}^3$,
- calcule $\log_{g_\gamma}(w).e_\alpha(x, y)$ dans M_δ muni de ses lois $+_\delta$ et $-_\delta$.

Un algorithme résolvant le problème décisionnel Diffie-Hellman bilinéaire :

- reçoit en entrée un paramètre $\alpha \in \Omega$, et $(w, x, y, z) \in {L_\gamma}^3 \times M_\delta$,
- décide si $z = \log_{g_\gamma}(w).e_\alpha(x, y)$ dans M_δ muni de ses lois $+_\delta$ et $-_\delta$.

Problèmes standards sur une famille avec couplage

On considère une famille de représentations de groupes cycliques avec couplages sur L et M.

Un algorithme résolvant le problème Diffie-Hellman bilinéaire :

- reçoit en entrée un paramètre $\alpha \in \Omega$, un triplet $(w, x, y) \in {L_\gamma}^3$,
- calcule $\log_{g_\gamma}(w).e_\alpha(x, y)$ dans M_δ muni de ses lois $+_\delta$ et $-_\delta$.

Un algorithme résolvant le problème décisionnel Diffie-Hellman bilinéaire :

- reçoit en entrée un paramètre $\alpha \in \Omega$, et $(w, x, y, z) \in {L_\gamma}^3 \times M_\delta$,
- décide si $z = \log_{g_\gamma}(w).e_\alpha(x, y)$ dans M_δ muni de ses lois $+_\delta$ et $-_\delta$.

Problèmes standards sur la famille générique avec couplage

On se place désormais dans le cadre de la famille générique de groupes cycliques avec couplage.

Un algorithme résolvant le problème Diffie-Hellman bilinéaire :

- reçoit en entrée une taille $n \in \mathbb{N}^*$, et $(0_f, g_f, 0_h, g_h, w, x, y) \in B(n)^7$,
- a accès à des oracles $+_f$, $-_f$, $+_h$, $-_h$, et $e_{f,h}$,
- calcule $\log_{g_\gamma}(w).e_\alpha(x, y)$ dans $B(n)$ muni des lois $+_h$ et $-_h$.

Un algorithme résolvant le problème décisionnel Diffie-Hellman bilinéaire :

- reçoit en entrée $n \in \mathbb{N}^*$, et $(0_f, g_f, 0_h, g_h, w, x, y, z) \in B(n)^8$,
- a accès à des oracles $+_f$, $-_f$, $+_h$, $-_h$, et $e_{f,h}$,
- décide si $z = \log_{g_\gamma}(w).e_\alpha(x, y)$ dans $B(n)$ muni des lois $+_h$ et $-_h$.

Problèmes standards sur la famille générique avec couplage

On se place désormais dans le cadre de la famille générique de groupes cycliques avec couplage.

Un algorithme résolvant le problème Diffie-Hellman bilinéaire :

- reçoit en entrée une taille $n \in \mathbb{N}^*$, et $(0_f, g_f, 0_h, g_h, w, x, y) \in B(n)^7$,
- a accès à des oracles $+_f$, $-_f$, $+_h$, $-_h$, et $e_{f,h}$,
- calcule $\log_{g_\gamma}(w).e_\alpha(x, y)$ dans $B(n)$ muni des lois $+_h$ et $-_h$.

Un algorithme résolvant le problème décisionnel Diffie-Hellman bilinéaire :

- reçoit en entrée $n \in \mathbb{N}^*$, et $(0_f, g_f, 0_h, g_h, w, x, y, z) \in B(n)^8$,
- a accès à des oracles $+_f$, $-_f$, $+_h$, $-_h$, et $e_{f,h}$,
- décide si $z = \log_{g_\gamma}(w).e_\alpha(x, y)$ dans $B(n)$ muni des lois $+_h$ et $-_h$.

Problèmes standards sur la famille générique avec couplage

On se place désormais dans le cadre de la famille générique de groupes cycliques avec couplage.

Un algorithme résolvant le problème Diffie-Hellman bilinéaire :

- reçoit en entrée une taille $n \in \mathbb{N}^*$, et $(0_f, g_f, 0_h, g_h, w, x, y) \in B(n)^7$,
- a accès à des oracles $+_f$, $-_f$, $+_h$, $-_h$, et $e_{f,h}$,
- calcule $\log_{g_\gamma}(w).e_\alpha(x, y)$ dans $B(n)$ muni des lois $+_h$ et $-_h$.

Un algorithme résolvant le problème décisionnel Diffie-Hellman bilinéaire :

- reçoit en entrée $n \in \mathbb{N}^*$, et $(0_f, g_f, 0_h, g_h, w, x, y, z) \in B(n)^8$,
- a accès à des oracles $+_f$, $-_f$, $+_h$, $-_h$, et $e_{f,h}$,
- décide si $z = \log_{g_\gamma}(w).e_\alpha(x, y)$ dans $B(n)$ muni des lois $+_h$ et $-_h$.

Plan de l'exposé

Contexte (1)

On s'intéresse plus précisément au problème Diffie-Hellman bilinéaire dans le cadre de la famille générique de groupes cycliques avec couplage.

Un environnement $\mathcal{E}$ choisit $n \in \mathbb{N}^*$, deux bijections f et h de $\mathbb{Z}/n\mathbb{Z}$ dans $B(n)$, et un triplet $(x_1, x_2, x_3) \in B(n)^3$.

Un algorithme probabiliste $\mathcal{A}$, de capacité de calcul infinie, envoie des requêtes aux oracles $+_f$, $-_f$, $+_h$, $-_h$, et $e_{f,h}$, initialement à partir de $n \in \mathbb{N}^*$ et $(0_f, g_f, 0_h, g_h, x_1, x_2, x_3) \in B(n)^7$. Finalement, il répond par $z \in B(n)$.

On considère que $\mathcal{A}$ ne peut pas choisir d'éléments aléatoires dans $B(n)$. Il simule de tels choix à partir d'un tirage aléatoire dans $\mathbb{Z}/n\mathbb{Z}$ et de requêtes.

Contexte (1)

On s'intéresse plus précisément au problème Diffie-Hellman bilinéaire dans le cadre de la famille générique de groupes cycliques avec couplage.

Un environnement $\mathcal{E}$ choisit $n \in \mathbb{N}^*$, deux bijections f et h de $\mathbb{Z}/n\mathbb{Z}$ dans $B(n)$, et un triplet $(x_1, x_2, x_3) \in B(n)^3$.

Un algorithme probabiliste $\mathcal{A}$, de capacité de calcul infinie, envoie des requêtes aux oracles $+_f$, $-_f$, $+_h$, $-_h$, et $e_{f,h}$, initialement à partir de $n \in \mathbb{N}^*$ et $(0_f, g_f, 0_h, g_h, x_1, x_2, x_3) \in B(n)^7$. Finalement, il répond par $z \in B(n)$.

On considère que $\mathcal{A}$ ne peut pas choisir d'éléments aléatoires dans $B(n)$. Il simule de tels choix à partir d'un tirage aléatoire dans $\mathbb{Z}/n\mathbb{Z}$ et de requêtes.

Contexte (1)

On s'intéresse plus précisément au problème Diffie-Hellman bilinéaire dans le cadre de la famille générique de groupes cycliques avec couplage.

Un environnement $\mathcal{E}$ choisit $n \in \mathbb{N}^*$, deux bijections f et h de $\mathbb{Z}/n\mathbb{Z}$ dans $B(n)$, et un triplet $(x_1, x_2, x_3) \in B(n)^3$.

Un algorithme probabiliste $\mathcal{A}$, de capacité de calcul infinie, envoie des requêtes aux oracles $+_f$, $-_f$, $+_h$, $-_h$, et $e_{f,h}$, initialement à partir de $n \in \mathbb{N}^*$ et $(0_f, g_f, 0_h, g_h, x_1, x_2, x_3) \in B(n)^7$. Finalement, il répond par $z \in B(n)$.

On considère que $\mathcal{A}$ ne peut pas choisir d'éléments aléatoires dans $B(n)$. Il simule de tels choix à partir d'un tirage aléatoire dans $\mathbb{Z}/n\mathbb{Z}$ et de requêtes.

Contexte (1)

On s'intéresse plus précisément au problème Diffie-Hellman bilinéaire dans le cadre de la famille générique de groupes cycliques avec couplage.

Un environnement $\mathcal{E}$ choisit $n \in \mathbb{N}^*$, deux bijections f et h de $\mathbb{Z}/n\mathbb{Z}$ dans $B(n)$, et un triplet $(x_1, x_2, x_3) \in B(n)^3$.

Un algorithme probabiliste $\mathcal{A}$, de capacité de calcul infinie, envoie des requêtes aux oracles $+_f$, $-_f$, $+_h$, $-_h$, et $e_{f,h}$, initialement à partir de $n \in \mathbb{N}^*$ et $(0_f, g_f, 0_h, g_h, x_1, x_2, x_3) \in B(n)^7$. Finalement, il répond par $z \in B(n)$.

On considère que $\mathcal{A}$ ne peut pas choisir d'éléments aléatoires dans $B(n)$. Il simule de tels choix à partir d'un tirage aléatoire dans $\mathbb{Z}/n\mathbb{Z}$ et de requêtes.

Contexte (2)

L'information obtenue progresivement par $\mathcal{A}$ peut être modélisée par deux suites de listes dans $B(n) \times (\mathbb{Z}/n\mathbb{Z})[X_1, X_2, X_3]$: R pour l'information sur le premier groupe, et S pour l'information sur le second groupe.

- $R_0 = \{(0_f, 0), (g_f, 1), (x_1, X_1), (x_2, X_2), (x_3, X_3)\}$,
- Si la $k^{\text{ième}}$ requête est : $a +_f b = c$, alors $R_k = R_{k-1} \cup (c, P_a + P_b)$,
- Si la $k^{\text{ième}}$ requête est : $-_f a = c$, alors $R_k = R_{k-1} \cup (c, -P_a)$.

- $S_0 = \{(0_h, 0), (g_h, 1)\}$,
- Si la $k^{\text{ième}}$ requête est : $a +_h b = c$, alors $S_k = S_{k-1} \cup (c, Q_a + Q_b)$,
- Si la $k^{\text{ième}}$ requête est : $-_h a = c$, alors $S_k = S_{k-1} \cup (c, -Q_a)$,
- Si la $k^{\text{ième}}$ requête est : $e_{f,h}(a, b) = c$, alors $S_k = S_{k-1} \cup (c, P_a.P_b)$.

Contexte (2)

L'information obtenue progresivement par $\mathcal{A}$ peut être modélisée par deux suites de listes dans $B(n) \times (\mathbb{Z}/n\mathbb{Z})[X_1, X_2, X_3]$: R pour l'information sur le premier groupe, et S pour l'information sur le second groupe.

- $\mathrm{R}_0 = \{(0_f, 0), (g_f, 1), (x_1, X_1), (x_2, X_2), (x_3, X_3)\}$,
- Si la $k^{\text{ième}}$ requête est : $a +_f b = c$, alors $\mathrm{R}_k = \mathrm{R}_{k-1} \cup (c, P_a + P_b)$,
- Si la $k^{\text{ième}}$ requête est : $-_f a = c$, alors $\mathrm{R}_k = \mathrm{R}_{k-1} \cup (c, -P_a)$.

- $\mathrm{S}_0 = \{(0_h, 0), (g_h, 1)\}$,
- Si la $k^{\text{ième}}$ requête est : $a +_h b = c$, alors $\mathrm{S}_k = \mathrm{S}_{k-1} \cup (c, Q_a + Q_b)$,
- Si la $k^{\text{ième}}$ requête est : $-_h a = c$, alors $\mathrm{S}_k = \mathrm{S}_{k-1} \cup (c, -Q_a)$,
- Si la $k^{\text{ième}}$ requête est : $e_{f,h}(a, b) = c$, alors $\mathrm{S}_k = \mathrm{S}_{k-1} \cup (c, P_a.P_b)$.

Contexte (2)

L'information obtenue progresivement par $\mathcal{A}$ peut être modélisée par deux suites de listes dans $B(n) \times (\mathbb{Z}/n\mathbb{Z})[X_1, X_2, X_3]$: R pour l'information sur le premier groupe, et S pour l'information sur le second groupe.

- $\mathrm{R}_0 = \{(0_f, 0), (g_f, 1), (x_1, X_1), (x_2, X_2), (x_3, X_3)\}$,
- Si la $k^{\text{ième}}$ requête est : $a +_f b = c$, alors $\mathrm{R}_k = \mathrm{R}_{k-1} \cup (c, P_a + P_b)$,
- Si la $k^{\text{ième}}$ requête est : $-_f a = c$, alors $\mathrm{R}_k = \mathrm{R}_{k-1} \cup (c, -P_a)$.

- $\mathrm{S}_0 = \{(0_h, 0), (g_h, 1)\}$,
- Si la $k^{\text{ième}}$ requête est : $a +_h b = c$, alors $\mathrm{S}_k = \mathrm{S}_{k-1} \cup (c, Q_a + Q_b)$,
- Si la $k^{\text{ième}}$ requête est : $-_h a = c$, alors $\mathrm{S}_k = \mathrm{S}_{k-1} \cup (c, -Q_a)$,
- Si la $k^{\text{ième}}$ requête est : $e_{f,h}(a, b) = c$, alors $\mathrm{S}_k = \mathrm{S}_{k-1} \cup (c, P_a.P_b)$.

Collision, cohérence et compatibilité

Il y a collision dans (R_k, S_k) si on peut trouver deux éléments ayant des polynômes différents et des valeurs identiques, soit dans R_k, soit dans S_k.

Un triplet $(\alpha_1, \alpha_2, \alpha_3) \in (\mathbb{Z}/n\mathbb{Z})^3$ est cohérent avec un ensemble de polynômes $\mathcal{P} \subset (\mathbb{Z}/n\mathbb{Z})[X_1, X_2, X_3]$ si :

$$\forall (P, P') \in \mathcal{P}^2, P \neq P' \Longrightarrow (P' - P)(\alpha_1, \alpha_2, \alpha_3) \neq 0\,.$$

Un couple de bijections (f, h) de $\mathbb{Z}/n\mathbb{Z}$ dans $B(n)$ est compatible avec (R_k, S_k) si :

$$\begin{cases} \forall (v, P) \in R_k,\ f(P(f^{-1}(x_1), f^{-1}(x_2), f^{-1}(x_3))) = v, \\ \forall (v, Q) \in S_k,\ h(Q(f^{-1}(x_1), f^{-1}(x_2), f^{-1}(x_3))) = v. \end{cases}$$

Collision, cohérence et compatibilité

Il y a collision dans (R_k, S_k) si on peut trouver deux éléments ayant des polynômes différents et des valeurs identiques, soit dans R_k, soit dans S_k.

Un triplet $(\alpha_1, \alpha_2, \alpha_3) \in (\mathbb{Z}/n\mathbb{Z})^3$ est cohérent avec un ensemble de polynômes $\mathcal{P} \subset (\mathbb{Z}/n\mathbb{Z})[X_1, X_2, X_3]$ si :

$$\forall (P, P') \in \mathcal{P}^2, P \neq P' \Longrightarrow (P' - P)(\alpha_1, \alpha_2, \alpha_3) \neq 0\,.$$

Un couple de bijections (f, h) de $\mathbb{Z}/n\mathbb{Z}$ dans $B(n)$ est compatible avec (R_k, S_k) si :

$$\begin{cases} \forall (v, P) \in R_k,\ f(P(f^{-1}(x_1), f^{-1}(x_2), f^{-1}(x_3))) = v, \\ \forall (v, Q) \in S_k,\ h(Q(f^{-1}(x_1), f^{-1}(x_2), f^{-1}(x_3))) = v. \end{cases}$$

Collision, cohérence et compatibilité

Il y a collision dans (R_k, S_k) si on peut trouver deux éléments ayant des polynômes différents et des valeurs identiques, soit dans R_k, soit dans S_k.

Un triplet $(\alpha_1, \alpha_2, \alpha_3) \in (\mathbb{Z}/n\mathbb{Z})^3$ est cohérent avec un ensemble de polynômes $\mathcal{P} \subset (\mathbb{Z}/n\mathbb{Z})[X_1, X_2, X_3]$ si :

$$\forall (P, P') \in \mathcal{P}^2, P \neq P' \Longrightarrow (P' - P)(\alpha_1, \alpha_2, \alpha_3) \neq 0\,.$$

Un couple de bijections (f, h) de $\mathbb{Z}/n\mathbb{Z}$ dans $B(n)$ est compatible avec (R_k, S_k) si :

$$\begin{cases} \forall (v, P) \in R_k,\ f(P(f^{-1}(x_1), f^{-1}(x_2), f^{-1}(x_3))) = v, \\ \forall (v, Q) \in S_k,\ h(Q(f^{-1}(x_1), f^{-1}(x_2), f^{-1}(x_3))) = v. \end{cases}$$

Plan de l'exposé

Lemme fondammental

Lemme (Première collision sur requête)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$, $(\mathrm{R}_k, \mathrm{S}_k)$ *sans collision, et une* $(k+1)^{ème}$ *requête,*

- *si les polynômes dans* R_k *et* S_k *sont de degrés bornés par* d,
- *si le polynôme associé à la nouvelle requête est de degré borné par* d,
- *si* $\#\mathrm{R}_k + \#\mathrm{S}_k < \sqrt{2p/d}$,

alors la probabilité, sur l'ensemble des couples de bijections compatibles avec $(\mathrm{R}_k, \mathrm{S}_k)$, *que la nouvelle requête conduise à une collision dans* $(\mathrm{R}_{k+1}, \mathrm{S}_{k+1})$ *est majorée par :*

$$\frac{d(\#\mathrm{R}_k + \#\mathrm{S}_k)}{p - d(\#\mathrm{R}_k + \#\mathrm{S}_k)^2/2}.$$

Lemme fondammental

Lemme (Première collision sur requête)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$, $(\mathrm{R}_k, \mathrm{S}_k)$ *sans collision, et une* $(k+1)^{ème}$ *requête,*

- *si les polynômes dans* R_k *et* S_k *sont de degrés bornés par* d,
- *si le polynôme associé à la nouvelle requête est de degré borné par* d,
- *si* $\#\mathrm{R}_k + \#\mathrm{S}_k < \sqrt{2p/d}$,

alors la probabilité, sur l'ensemble des couples de bijections compatibles avec $(\mathrm{R}_k, \mathrm{S}_k)$, *que la nouvelle requête conduise à une collision dans* $(\mathrm{R}_{k+1}, \mathrm{S}_{k+1})$ *est majorée par :*

$$\frac{d(\#\mathrm{R}_k + \#\mathrm{S}_k)}{p - d(\#\mathrm{R}_k + \#\mathrm{S}_k)^2/2}.$$

Lemme fondammental

Lemme (Première collision sur requête)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$, $(\mathrm{R}_k, \mathrm{S}_k)$ *sans collision, et une* $(k+1)^{ème}$ *requête,*

- *si les polynômes dans* R_k *et* S_k *sont de degrés bornés par* d,
- *si le polynôme associé à la nouvelle requête est de degré borné par* d,
- *si* $\#\mathrm{R}_k + \#\mathrm{S}_k < \sqrt{2p/d}$,

alors la probabilité, sur l'ensemble des couples de bijections compatibles avec $(\mathrm{R}_k, \mathrm{S}_k)$, *que la nouvelle requête conduise à une collision dans* $(\mathrm{R}_{k+1}, \mathrm{S}_{k+1})$ *est majorée par :*

$$\frac{d(\#\mathrm{R}_k + \#\mathrm{S}_k)}{p - d(\#\mathrm{R}_k + \#\mathrm{S}_k)^2/2}.$$

Lemme fondammental

Lemme (Première collision sur requête)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$, $(\mathrm{R}_k, \mathrm{S}_k)$ *sans collision, et une* $(k+1)^{ème}$ *requête,*

- *si les polynômes dans* R_k *et* S_k *sont de degrés bornés par* d,
- *si le polynôme associé à la nouvelle requête est de degré borné par* d,
- *si* $\#\mathrm{R}_k + \#\mathrm{S}_k < \sqrt{2p/d}$,

alors la probabilité, sur l'ensemble des couples de bijections compatibles avec $(\mathrm{R}_k, \mathrm{S}_k)$, *que la nouvelle requête conduise à une collision dans* $(\mathrm{R}_{k+1}, \mathrm{S}_{k+1})$ *est majorée par :*

$$\frac{d(\#\mathrm{R}_k + \#\mathrm{S}_k)}{p - d(\#\mathrm{R}_k + \#\mathrm{S}_k)^2/2}.$$

Lemme fondammental

Lemme (Première collision sur requête)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$, $(\mathrm{R}_k, \mathrm{S}_k)$ *sans collision, et une* $(k+1)^{\text{ème}}$ *requête,*

- *si les polynômes dans* R_k *et* S_k *sont de degrés bornés par* d,
- *si le polynôme associé à la nouvelle requête est de degré borné par* d,
- *si* $\#\mathrm{R}_k + \#\mathrm{S}_k < \sqrt{2p/d}$,

alors la probabilité, sur l'ensemble des couples de bijections compatibles avec $(\mathrm{R}_k, \mathrm{S}_k)$, *que la nouvelle requête conduise à une collision dans* $(\mathrm{R}_{k+1}, \mathrm{S}_{k+1})$ *est majorée par :*

$$\frac{d(\#\mathrm{R}_k + \#\mathrm{S}_k)}{p - d(\#\mathrm{R}_k + \#\mathrm{S}_k)^2/2}.$$

Existence d'une collision

Corollaire (Existence d'une collision)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$,

- *si les polynômes associés aux requêtes sont au plus quadratiques,*
- *si $k \leq \sqrt{p} - 7$,*

alors la probabilité, sur l'ensemble des couples de bijections compatibles avec (R_0, S_0), qu'une série de k requêtes conduise à une collision dans (R_k, S_k) est majorée par :

$$\frac{2(k+6)^2}{p-(k+6)^2}.$$

Existence d'une collision

Corollaire (Existence d'une collision)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$,

- *si les polynômes associés aux requêtes sont au plus quadratiques,*
- *si* $k \leq \sqrt{p} - 7$,

alors la probabilité, sur l'ensemble des couples de bijections compatibles avec (R_0, S_0), *qu'une série de* k *requêtes conduise à une collision dans* (R_k, S_k) *est majorée par :*

$$\frac{2(k+6)^2}{p-(k+6)^2}.$$

Existence d'une collision

Corollaire (Existence d'une collision)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$,

- *si les polynômes associés aux requêtes sont au plus quadratiques,*
- *si* $k \leq \sqrt{p} - 7$,

alors la probabilité, sur l'ensemble des couples de bijections compatibles avec (R_0, S_0), *qu'une série de* k *requêtes conduise à une collision dans* (R_k, S_k) *est majorée par :*

$$\frac{2(k+6)^2}{p-(k+6)^2}.$$

Existence d'une collision

Corollaire (Existence d'une collision)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$,

- *si les polynômes associés aux requêtes sont au plus quadratiques,*
- *si* $k \leq \sqrt{p} - 7$,

alors la probabilité, sur l'ensemble des couples de bijections compatibles avec (R_0, S_0), *qu'une série de* k *requêtes conduise à une collision dans* (R_k, S_k) *est majorée par :*

$$\frac{2(k+6)^2}{p-(k+6)^2}\,.$$

Succès d'un algorithme sans collision

Corollaire (Succès d'un algorithme sans collision)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$, $\mathcal{A}$ un algorithme résolvant le problème Diffie-Hellman bilinéaire dans la famille générique de groupes cycliques avec couplage,

- *si $k < \sqrt{2p/3} - 7$,*
- *si (R_k, S_k) est sans collision,*

alors la probabilité de succès de $\mathcal{A}$, après k requêtes aux oracles, est majorée par :

$$\frac{3(k+7)}{p - 3(k+7)^2/2}.$$

Succès d'un algorithme sans collision

Corollaire (Succès d'un algorithme sans collision)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$, $\mathcal{A}$ un algorithme résolvant le problème Diffie-Hellman bilinéaire dans la famille générique de groupes cycliques avec couplage,

- *si $k < \sqrt{2p/3} - 7$,*
- *si (R_k, S_k) est sans collision,*

alors la probabilité de succès de $\mathcal{A}$, après k requêtes aux oracles, est majorée par :

$$\frac{3(k+7)}{p - 3(k+7)^2/2}.$$

Succès d'un algorithme sans collision

Corollaire (Succès d'un algorithme sans collision)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$, $\mathcal{A}$ un algorithme résolvant le problème Diffie-Hellman bilinéaire dans la famille générique de groupes cycliques avec couplage,

- *si $k < \sqrt{2p/3} - 7$,*
- *si (R_k, S_k) est sans collision,*

alors la probabilité de succès de $\mathcal{A}$, après k requêtes aux oracles, est majorée par :

$$\frac{3(k+7)}{p - 3(k+7)^2/2}.$$

Succès d'un algorithme sans collision

Corollaire (Succès d'un algorithme sans collision)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$, $\mathcal{A}$ un algorithme résolvant le problème Diffie-Hellman bilinéaire dans la famille générique de groupes cycliques avec couplage,

- *si $k < \sqrt{2p/3} - 7$,*
- *si (R_k, S_k) est sans collision,*

alors la probabilité de succès de $\mathcal{A}$, après k requêtes aux oracles, est majorée par :

$$\frac{3(k+7)}{p - 3(k+7)^2/2}.$$

Théorème complet

Théorème (Difficulté du problème Diffie-Hellman bilinéaire)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$, $\mathcal{A}$ un algorithme résolvant le problème Diffie-Hellman bilinéaire dans la famille générique de groupes cycliques avec couplage, si $k \leq \sqrt{p} - 7$, alors la probabilité de succès de $\mathcal{A}$, après k requêtes aux oracles, est majorée par :

$$\frac{2(k+6)^2}{p-(k+6)^2} + \frac{3(k+7)}{p-3(k+7)^2/2} = O(k^2/p)\,.$$

Pour un ordre quelconque, le plus grand diviseur premier de n joue le rôle de p (via une réduction au cas précédent).

Théorème complet

Théorème (Difficulté du problème Diffie-Hellman bilinéaire)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$, $\mathcal{A}$ un algorithme résolvant le problème Diffie-Hellman bilinéaire dans la famille générique de groupes cycliques avec couplage, si $k \leq \sqrt{p} - 7$, alors la probabilité de succès de $\mathcal{A}$, après k requêtes aux oracles, est majorée par :

$$\frac{2(k+6)^2}{p-(k+6)^2} + \frac{3(k+7)}{p-3(k+7)^2/2} = O(k^2/p)\,.$$

Pour un ordre quelconque, le plus grand diviseur premier de n joue le rôle de p (via une réduction au cas précédent).

Théorème complet

Théorème (Difficulté du problème Diffie-Hellman bilinéaire)

Soit $n = p^\lambda$, $k \in \mathbb{N}^*$, $\mathcal{A}$ un algorithme résolvant le problème Diffie-Hellman bilinéaire dans la famille générique de groupes cycliques avec couplage, si $k \leq \sqrt{p} - 7$, alors la probabilité de succès de $\mathcal{A}$, après k requêtes aux oracles, est majorée par :

$$\frac{2(k+6)^2}{p-(k+6)^2} + \frac{3(k+7)}{p-3(k+7)^2/2} = O(k^2/p)\,.$$

Pour un ordre quelconque, le plus grand diviseur premier de n joue le rôle de p (via une réduction au cas précédent).

Conclusion

Contribution :

- On a donné une formalisation précise des familles génériques de groupes cycliques avec et sans couplage,
- On a prouvé la difficulté du problème Diffie-Hellman bilinéaire dans le cadre de la famille générique de groupes cycliques avec couplage.

Perspectives :

- Les problèmes décisionnels (bilinéaire et non-bilinéaire) sont-ils, eux aussi, difficiles ?
- Le modèle de la famille générique de groupes cycliques avec couplage peut-elle être utilisée directement pour prouver des protocoles utilisant les couplages ?

Des Questions ?

Conclusion

Contribution :

- On a donné une formalisation précise des familles génériques de groupes cycliques avec et sans couplage,
- On a prouvé la difficulté du problème Diffie-Hellman bilinéaire dans le cadre de la famille générique de groupes cycliques avec couplage.

Perspectives :

- Les problèmes décisionnels (bilinéaire et non-bilinéaire) sont-ils, eux aussi, difficiles ?
- Le modèle de la famille générique de groupes cycliques avec couplage peut-elle être utilisée directement pour prouver des protocoles utilisant les couplages ?

Des Questions ?

Conclusion

Contribution :

- On a donné une formalisation précise des familles génériques de groupes cycliques avec et sans couplage,
- On a prouvé la difficulté du problème Diffie-Hellman bilinéaire dans le cadre de la famille générique de groupes cycliques avec couplage.

Perspectives :

- Les problèmes décisionnels (bilinéaire et non-bilinéaire) sont-ils, eux aussi, difficiles ?
- Le modèle de la famille générique de groupes cycliques avec couplage peut-elle être utilisée directement pour prouver des protocoles utilisant les couplages ?

Des Questions ?

Conclusion

Contribution :

- On a donné une formalisation précise des familles génériques de groupes cycliques avec et sans couplage,
- On a prouvé la difficulté du problème Diffie-Hellman bilinéaire dans le cadre de la famille générique de groupes cycliques avec couplage.

Perspectives :

- Les problèmes décisionnels (bilinéaire et non-bilinéaire) sont-ils, eux aussi, difficiles ?
- Le modèle de la famille générique de groupes cycliques avec couplage peut-elle être utilisée directement pour prouver des protocoles utilisant les couplages ?

Des Questions ?

Conclusion

Contribution :

- On a donné une formalisation précise des familles génériques de groupes cycliques avec et sans couplage,
- On a prouvé la difficulté du problème Diffie-Hellman bilinéaire dans le cadre de la famille générique de groupes cycliques avec couplage.

Perspectives :

- Les problèmes décisionnels (bilinéaire et non-bilinéaire) sont-ils, eux aussi, difficiles ?
- Le modèle de la famille générique de groupes cycliques avec couplage peut-elle être utilisée directement pour prouver des protocoles utilisant les couplages ?

Des Questions ?

Conclusion

Contribution :

- On a donné une formalisation précise des familles génériques de groupes cycliques avec et sans couplage,
- On a prouvé la difficulté du problème Diffie-Hellman bilinéaire dans le cadre de la famille générique de groupes cycliques avec couplage.

Perspectives :

- Les problèmes décisionnels (bilinéaire et non-bilinéaire) sont-ils, eux aussi, difficiles ?
- Le modèle de la famille générique de groupes cycliques avec couplage peut-elle être utilisée directement pour prouver des protocoles utilisant les couplages ?

Des Questions ?

Conclusion

Contribution :

- On a donné une formalisation précise des familles génériques de groupes cycliques avec et sans couplage,
- On a prouvé la difficulté du problème Diffie-Hellman bilinéaire dans le cadre de la famille générique de groupes cycliques avec couplage.

Perspectives :

- Les problèmes décisionnels (bilinéaire et non-bilinéaire) sont-ils, eux aussi, difficiles ?
- Le modèle de la famille générique de groupes cycliques avec couplage peut-elle être utilisée directement pour prouver des protocoles utilisant les couplages ?

Des Questions ?