Cryptography via Burnside Groups

Nelly Fazio
City College of CUNY

Antonio R. Nicolosi
Stevens Institute of Technology

Based on joint work with G. Baumslag, K. Iga, L. Perret, V. Shpilrain and W.E. Skeith III
Goal

Seek sources of viable intractability assumptions from combinatorial group theory

- Cryptographically useful
- Evidence of (average-case) hardness (random self-reducibility)

Approach

- Generalize well-established crypto assumptions (LPN/LWE) to a group-theoretic setting
- Study instantiation in suitable non-commutative groups
In Memoriam

Gilbert Baumslag (1933–2014)
1 Background
- Burnside Groups ($B_n$)
- Learning Burnside Homomorphisms with Noise ($B_n$-LHN)

2 Random Self-Reducibility of $B_n$-LHN

3 Cryptography via Burnside Groups
- Minicrypt via Burnside Groups
- Cryptomania via Burnside Groups? (future work)
1 Background
   - Burnside Groups ($B_n$)
   - Learning Burnside Homomorphisms with Noise ($B_n$-LHN)

2 Random Self-Reducibility of $B_n$-LHN

3 Cryptography via Burnside Groups
   - Minicrypt via Burnside Groups
   - Cryptomania via Burnside Groups? (future work)
Burnside Problem (Informal)

- Are groups whose elements all have finite order necessarily finite?
- What is their combinatorial structure?
\[ B(n, m) \]: “Most generic” group with \( n \) generators where the order of all elements divides \( m \)

- Generators \( x_1, \ldots, x_n \) (like indeterminates in a multivariate poly)
- Elements are sequences of \( x_i \) and \( x_i^{-1} \)
- Empty sequence is the identity element of the group
- Exponent condition: For every \( w \in B(n, m) \) it holds that \( w^m = 1 \)
Burnside group of exponent $m$

- $B(n, m)$: “Most generic” group with $n$ generators where the order of all elements divides $m$
  - Generators $x_1, \ldots, x_n$ (like indeterminates in a multivariate poly)
  - Elements are sequences of $x_i$ and $x_i^{-1}$
  - Empty sequence is the identity element of the group
  - Exponent condition: For every $w \in B(n, m)$ it holds that $w^m = 1$
**$B(n, m)$**: “Most generic” group with $n$ generators where the order of all elements divides $m$

- Generators $x_1, \ldots, x_n$ (like indeterminates in a multivariate poly)
- Elements are sequences of $x_i$ and $x_i^{-1}$
- Empty sequence is the identity element of the group
- Exponent condition: For every $w \in B(n, m)$ it holds that $w^m = 1$
$B(n, m)$: “Most generic” group with $n$ generators where the order of all elements divides $m$

- Generators $x_1, \ldots, x_n$ (like indeterminates in a multivariate poly)
- Elements are sequences of $x_i$ and $x_i^{-1}$
- Empty sequence is the identity element of the group

Exponent condition: For every $w \in B(n, m)$ it holds that $w^m = 1$
Burnside group of exponent $m$

- $B(n, m)$: “Most generic” group with $n$ generators where the order of all elements divides $m$
  - Generators $x_1, \ldots, x_n$ (like indeterminates in a multivariate poly)
  - Elements are sequences of $x_i$ and $x_i^{-1}$
  - Empty sequence is the identity element of the group
  - Exponent condition: For every $w \in B(n, m)$ it holds that $w^m = 1$
Characterizing $B(n, m)$ not so easy . . .

| $B(n, 2)$  | Finite and abelian, isomorphic to $(\mathbb{F}_2^n, +)$ |
| $B(n, 3)$  | Finite, non-commutative, much larger than $(\mathbb{F}_3^n, +)$ |
| $B(n, 4)$  | Finite |
| $B(n, 5)$  | Unknown |
| $B(n, 6)$  | Finite |
| $B(n, 7)$  | Unknown |
| $B(n, m)$, $m$ “large” | Infinite |

Will focus on $B(n, 3)$ (simplest case beyond vector spaces)

Notation: $B_n \doteq B(n, 3)$
Characterizing $B(n, m)$ not so easy . . .

- $B(n, 2)$ Finite and abelian, isomorphic to $(\mathbb{F}_2^n, +)$
- $B(n, 3)$ Finite, non-commutative, much larger than $(\mathbb{F}_3^n, +)$
- $B(n, 4)$ Finite
- $B(n, 5)$ Unknown
- $B(n, 6)$ Finite
- $B(n, 7)$ Unknown

... $B(n, m)$, $m$ “large” Infinite

Will focus on $B(n, 3)$ (simplest case beyond vector spaces)

- Notation: $B_n \doteq B(n, 3)$
\( B_n \): “Most generic” group with \( n \) generators where the order of all non-identity elements is 3

- Generators \( x_1, \ldots, x_n \)
- Elements are sequences of \( x_i \) and \( x_i^{-1} \)
- Exponent condition: \( \forall w \in B_n, w w w = 1 \) (\( \ast \))

Q: “Most generic”!? 
A: The only non-trivial identities in \( B_n \) are those implied by (\( \ast \))

\( \Rightarrow \) \( B_n \) non-commutative

- \( x_i x_j \neq x_j x_i \) for any two distinct generators \( (i \neq j) \)

\( \Rightarrow \) Group operation in \( B_n \) defined “formally”

- To “multiply” \( w_1, w_2 \in B_n \), just concatenate them
- Simplifications may arise at the interface of \( w_1 \) and \( w_2 \)
$B_n$: “Most generic” group with $n$ generators where the order of all non-identity elements is 3

- Generators $x_1, \ldots, x_n$
- Elements are sequences of $x_i$ and $x_i^{-1}$
- Exponent condition: $\forall w \in B_n$, $www = 1$ (\star)

Q: “Most generic”!?  
A: The only non-trivial identities in $B_n$ are those implied by (\star)

$\Rightarrow$ $B_n$ non-commutative

- $x_ix_j \neq x_jx_i$ for any two distinct generators ($i \neq j$)

$\Rightarrow$ Group operation in $B_n$ defined “formally”

- To “multiply” $w_1, w_2 \in B_n$, just concatenate them
- Simplifications may arise at the interface of $w_1$ and $w_2$
\( B_n: \) Burnside Groups of Exponent 3

- \( B_n: \) "Most generic" group with \( n \) generators where the order of all non-identity elements is 3
  - Generators \( x_1, \ldots, x_n \)
  - Elements are sequences of \( x_i \) and \( x_i^{-1} \)
  - Exponent condition: \( \forall w \in B_n, \quad www = 1 \) (\( \star \))

- Q: "Most generic"!?
  - A: The only non-trivial identities in \( B_n \) are those implied by (\( \star \))
  - \( B_n \) non-commutative
    - \( x_i x_j \neq x_j x_i \) for any two distinct generators (\( i \neq j \))
  - Group operation in \( B_n \) defined "formally"
    - To "multiply" \( w_1, w_2 \in B_n \), just concatenate them
    - Simplifications may arise at the interface of \( w_1 \) and \( w_2 \)
Burnside Groups of Exponent 3

- $B_n$: “Most generic” group with $n$ generators where the order of all non-identity elements is 3
  - Generators $x_1, \ldots, x_n$
  - Elements are sequences of $x_i$ and $x_i^{-1}$
  - Exponent condition: $\forall w \in B_n, www = 1$ (*)

- Q: “Most generic”!?
  - A: The only non-trivial identities in $B_n$ are those implied by (*)

- $B_n$ non-commutative
  - $x_i x_j \neq x_j x_i$ for any two distinct generators ($i \neq j$)

- Group operation in $B_n$ defined “formally”
  - To “multiply” $w_1, w_2 \in B_n$, just concatenate them
  - Simplifications may arise at the interface of $w_1$ and $w_2$
In $B_n$, $x_ix_j \neq x_jx_i$ for any two distinct generators ($i \neq j$).

However, always possible to get $x_ix_j = x_jx_i[x_i, x_j]$ by defining

$$[x_i, x_j] = x_i^{-1}x_j^{-1}x_ix_j$$

Call $[x_i, x_j]$ a 2-commutator.

Similarly, define a 3-commutator $[x_i, x_j, x_k]$ as

$$[x_i, x_j, x_k] = [[x_i, x_j], x_k]$$

In general, may define $\ell$-commutators inductively, but in $B_n$ all $\ell$-commutators vanish for $\ell \geq 4$,

$$[x_i, x_j, x_k, x_h] = 1$$
Commutators

- In $B_n$, $x_ix_j \neq x_jx_i$ for any two distinct generators ($i \neq j$)
- However, always possible to get $x_ix_j = x_jx_i[x_i, x_j]$ by defining

$$[x_i, x_j] = x_i^{-1}x_j^{-1}x_ix_j$$

Call $[x_i, x_j]$ a 2-commutator
- Similarly, define a 3-commutator $[x_i, x_j, x_k]$ as

$$[x_i, x_j, x_k] = [[x_i, x_j], x_k]$$

- In general, may define $\ell$-commutators inductively, but in $B_n$ all $\ell$-commutators vanish for $\ell \geq 4$,

$$[x_i, x_j, x_k, x_h] = 1$$
Commutators

- In $B_n$, $x_i x_j \neq x_j x_i$ for any two distinct generators ($i \neq j$)
- However, always possible to get $x_i x_j = x_j x_i [x_i, x_j]$ by defining

\[ [x_i, x_j] = x_i^{-1} x_j^{-1} x_i x_j \]

Call $[x_i, x_j]$ a **2-commutator**

- Similarly, define a **3-commutator** $[x_i, x_j, x_k]$ as

\[ [x_i, x_j, x_k] = [[[x_i, x_j], x_k] \]

- In general, may define **$\ell$-commutators** inductively, but in $B_n$ all $\ell$-commutators vanish for $\ell \geq 4$,

\[ [x_i, x_j, x_k, x_h] = 1 \]
Commutators Identities in $B_n$

- $[x_i, x_j, x_k, x_h] = 1$ implies:
  - 3-commutators commute with all $w \in B_n$:
    \[
    [x_i, x_j, x_k]w = w[x_i, x_j, x_k]
    \]
  - 2-commutators commute among themselves:
    \[
    [x_k, x_h][x_i, x_j] = [x_i, x_j][x_k, x_h]
    \]
- Other commutator identities in $B_n$:
  \[
  [x_j, x_i] = [x_i, x_j]^{-1} = [x_i, x_j^{-1}] = [x_j^{-1}, x_i] \quad [x_i, x_j, x_i] = 1
  \]
  \[
  [x_i, x_j, x_k] = [x_k, x_j, x_i]^{-1} \quad [x_i, x_j, x_k] = [x_j, x_k, x_i] = [x_k, x_i, x_j]
  \]

[upshot: w.l.o.g, generators always sorted within commutator]
Commutators Identities in $B_n$

- $[x_i, x_j, x_k, x_h] = 1$ implies:
  - 3-commutators commute with all $w \in B_n$:
    
    $$[x_i, x_j, x_k]w = w[x_i, x_j, x_k]$$

- 2-commutators commute among themselves:
  
  $$[x_k, x_h][x_i, x_j] = [x_i, x_j][x_k, x_h]$$

- Other commutator identities in $B_n$:
  
  $$[x_j, x_i] = [x_i, x_j]^{-1} = [x_i, x_j^{-1}] = [x_i^{-1}, x_j]$$
  $$[x_i, x_j, x_k] = [x_k, x_j, x_i]^{-1} = [x_j, x_k, x_i] = [x_k, x_i, x_j]$$

[upshot: w.l.o.g, generators always sorted within commutator]
Commutators Identities in $B_n$

- $[x_i, x_j, x_k, x_h] = 1$ implies:
  - 3-commutators commute with all $w \in B_n$:
    $$[x_i, x_j, x_k]w = w[x_i, x_j, x_k]$$
  - 2-commutators commute among themselves:
    $$[x_k, x_h][x_i, x_j] = [x_i, x_j][x_k, x_h]$$

- Other commutator identities in $B_n$:
  $$[x_j, x_i] = [x_i, x_j]^{-1} = [x_i, x_j^{-1}] = [x_i^{-1}, x_j] \quad [x_i, x_j, x_i] = 1$$
  $$[x_i, x_j, x_k] = [x_k, x_j, x_i]^{-1} \quad [x_i, x_j, x_k] = [x_j, x_k, x_i] = [x_k, x_i, x_j]$$

*upshot: w.l.o.g, generators always sorted within commutator*
In general, elements in non-commutative groups may have multiple equivalent forms.

*E.g.*, in $B_n$

$$x_ix_j^{-1}x_i = x_jx_i^{-1}x_j$$

because

$$x_ix_j^{-1}x_i x_j^{-1} x_i x_j^{-1} = (x_i x_j^{-1})^3 = 1$$

In $B_n$, commutator identities imply that any $w \in B_n$ can always be written uniquely as:

$$w = \prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}}$$

where $\alpha_i, \beta_{i,j}, \gamma_{i,j,k} \in \{-1, 0, 1\}$, for all $1 \leq i < j < k \leq n$
In general, elements in non-commutative groups may have multiple equivalent forms

*E.g.*, in $B_n$

$$x_i x_j^{-1} x_i = x_j x_i^{-1} x_j$$

because

$$x_i x_j^{-1} x_i x_j^{-1} x_i x_j^{-1} = (x_i x_j^{-1})^3 = 1$$

In $B_n$, commutator identities imply that any $w \in B_n$ can always be written uniquely as:

$$w = \prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}}$$

where $\alpha_i, \beta_{i,j}, \gamma_{i,j,k} \in \{-1, 0, 1\}$, for all $1 \leq i < j < k \leq n$
Example: The Structure of $B_2$

- Cayley graph of $B_2$ (left): nodes $\equiv$ elements; edges $\equiv$ multiplication by a generator (green: $x_1$; purple: $x_2$)
- $B_2$ has 27 elements, of the form
  \[ x_1^{\alpha_1} x_2^{\alpha_2} [x_1, x_2]^{\beta_{1,2}}, \alpha_1, \alpha_2, \beta_{1,2} \in \mathbb{F}_3 \]
- Isomorphic to Heisenberg Group $H_1(\mathbb{F}_3)$:
  \[
  \begin{pmatrix}
  1 & \alpha_1 & \beta_{1,2} \\
  0 & 1 & \alpha_2 \\
  0 & 0 & 1
  \end{pmatrix} \in GL(3, \mathbb{F}_3)
  \]
- Beware of hasty generalization: for $n \geq 3$, $B_n \not\cong H_m(\mathbb{F}_3)$
- No known $poly(n)$-order representation of $B_n$
Recall the normal form in $B_n$:

$$\prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}}$$

To multiply two elements $w_1$ and $w_2$, first concatenate them . . .

. . . then reduce back to normal by reordering commutators via $O(n^3)$ three-stage collecting process
Recall the normal form in $B_n$:

$$\prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}}$$

$O(n)$  $O(n^2)$  $O(n^3)$

To multiply two elements $w_1$ and $w_2$, first concatenate them . . .

. . . then reduce back to normal by reordering commutators via $O(n^3)$ three-stage collecting process.
Recall the normal form in $B_n$:

$$
\prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}}
$$

To multiply two elements $w_1$ and $w_2$, first concatenate them . . .

. . . then reduce back to normal by reordering commutators via $O(n^3)$ three-stage collecting process.
Recall the normal form in $B_n$:

$$\prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}}$$

To multiply two elements $w_1$ and $w_2$, first concatenate them...

...then reduce back to normal by reordering commutators via $O(n^3)$ three-stage collecting process.
Recall the normal form in $B_n$:

$$\prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}}$$

- To multiply two elements $w_1$ and $w_2$, first concatenate them...

  - ... then reduce back to normal by reordering commutators via $O(n^3)$ three-stage `collecting process`
The Collecting Process (1/3)

Stage 1

Aggregate 3-commutators in $w_1$ and $w_2$, adding matching exponents mod 3

Time: $O(1)$ per 3-commutator, total $O(n^3)$
Stage 2
Move 2-commutators in $w_1$ to the right of generators in $w_2$

Each 2-commutator traveling right incurs $O(n)$ (constant-time) swaps with generators in $w_2$.

Time: $O(n)$ per 2-commutator, total $O(n^3)$
The Collecting Process (3/3)

Stage 3

Restore lexicographic order among generators

Fixing each out-of-order generator takes $O(n)$ swaps, and each swap creates a 2-commutator.

Before moving on to the next generator, these $O(n)$ 2-commutators must travel rightward (similarly to step 2 above), which takes $O(n^2)$ steps.

Time: $O(n^2)$ per generator, total $O(n^3)$
Burnside Groups: Recap

- Compact normal form:

\[
\prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}}
\]

\[|B_n| = 3^n + \binom{n}{2} + \binom{n}{3}\]

- Efficient \((O(n^3))\) group operation
  - Cubic in security parameter, but linear in input size
  - Similar (somewhat simpler) process to compute inverses (omitted)

- Non-commutative, but enjoys several useful identities
  - \(www = 1\) for any \(w \in B_n\)
  - \([x_i, x_j, x_k, x_h] = 1\) for any choice of generators

Q: What computational tasks are hard over Burnside groups?!
Burnside Groups: Recap

- Compact normal form:

\[
\prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}}
\]

\[\Rightarrow |B_n| = 3^n + \binom{n}{2} + \binom{n}{3}\]

- Efficient \((O(n^3))\) group operation
  - Cubic in security parameter, but linear in input size
  - Similar (somewhat simpler) process to compute inverses (omitted)
- Non-commutative, but enjoys several useful identities
  - \(www = 1\) for any \(w \in B_n\)
  - \([x_i, x_j, x_k, x_h] = 1\) for any choice of generators

Q: What computational tasks are hard over Burnside groups?!
Burnside Groups: Recap

- Compact normal form:

\[
\prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}}
\]

\[|B_n| = 3^{n+\binom{n}{2}+\binom{n}{3}}\]

- Efficient \((O(n^3))\) group operation
  - Cubic in security parameter, but linear in input size
  - Similar (somewhat simpler) process to compute inverses (omitted)

- Non-commutative, but enjoys several useful identities
  - \(www = 1\) for any \(w \in B_n\)
  - \([x_i, x_j, x_k, x_h] = 1\) for any choice of generators

Q: What computational tasks are hard over Burnside groups?!
Compact normal form:

\[
\prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}}
\]

\( \Rightarrow |B_n| = 3^n + \binom{n}{2} + \binom{n}{3} \)

Efficient (\(O(n^3)\)) group operation
  - Cubic in security parameter, but linear in input size
  - Similar (somewhat simpler) process to compute inverses (omitted)

Non-commutative, but enjoys several useful identities
  - \(www = 1\) for any \(w \in B_n\)
  - \([x_i, x_j, x_k, x_h] = 1\) for any choice of generators

Q: What computational tasks are hard over Burnside groups?!
Learning With Errors (LWE)

The LWE Setting

- $s \in \mathbb{F}_q^n$
- $\Psi_n$: a discrete gaussian distribution over $\mathbb{F}_q$ centered at 0
- $A_s^{\Psi_n}$: distribution on $\mathbb{F}_q^n \times \mathbb{F}_q$ whose samples are pairs $(a, b)$
  where $a \leftarrow \mathbb{F}_q^n, b = s \cdot a + e, e \leftarrow \Psi_n$

LWE Assumption

$$A_s^{\Psi_n} \approx_{\text{PPT}} \mathcal{U}(\mathbb{F}_q^n \times \mathbb{F}_q)$$
LWE over Groups: Learning Homomorphisms w/ Noise

Vector Spaces

\[ \mathbb{F}_q^n \ni a \approx s \cdot a \]
\[ \mathbb{F}_q \ni b = s \cdot a + e \]

Groups

\[ G_n \ni a \approx \varphi(a) \]
\[ P_n \ni b = \varphi(a)e \]

Learning With Errors

secret linear functional \( s \cdot \) 
“small” \( \mathbb{F}_q \)-noise \( e \)

Learning Homomorphisms w/ Noise

secret \( (G_n, P_n) \)-homomorphism \( \varphi \)
“small” \( P_n \)-noise \( e \)
The LHN Setting

- Groups $G_n$, $P_n$
- Distributions $\Gamma_n$, $\Psi_n$, $\Phi_n$ over $G_n$, $P_n$, $\text{hom}(G_n, P_n)$, resp.
- $A^\Psi_n(\varphi)$ (for $\varphi \in \text{hom}(G_n, P_n)$): Distribution over $G_n \times P_n$ whose samples are pairs $(a, b)$ where $a \leftarrow \Gamma_n$, $e \leftarrow \Psi_n$, $b = \varphi(a)e$

LHN Assumption

$$A^\Psi_n(\varphi) \approx_{\text{PPT}} U(G_n \times P_n), \quad \varphi \leftarrow \Phi_n$$
LWE As an Instance of LHN

- $G_n := (\mathbb{F}_p^n, +)$ and $\Gamma_n := U(\mathbb{F}_p^n)$
- $P_n := (\mathbb{F}_p, +)$ and $\psi_n := \text{discrete gaussian}$
- $\varphi := s \cdot _{}$ and $\Phi_n := U(\text{hom}(\mathbb{F}_p^n, \mathbb{F}_p))$

Diagram:

```
\[ \begin{array}{ccc}
\mathbb{F}_p^n & \ni & a \\
\downarrow & & \downarrow \\
\mathbb{F}_p & \ni & b \\
\| & & \| \\
\varphi(a) & & \varphi(a)e
\end{array} \]
```
$G_n := B_n, P_n := B_r$ (r small constant, e.g., $r = 4$)

$\Gamma_n := U(B_n)$

$\Phi_n := U(\text{hom}(B_n, B_r))$

$\Psi_n := \left[ v \leftarrow U(\mathbb{F}_3^r), \sigma \leftarrow S_r : \prod_{i=1}^r x_{\sigma(i)}^{v_i} \right] \quad (S_r: r\text{-permutations})$

(uni. dist. over $B_r$-elements of Cayley-norm $\leq r =: B_r$)

$B_n \xrightarrow{\approx} \varphi \leftarrow U(\text{hom}(B_n, B_r))$

$\varphi(a)e, \quad (e \leftarrow \Psi_n)$
\[ B_n \text{-LHN: Instantiating LHN over Burnside Groups} \]

- \( G_n := B_n, \ P_n := B_r \) (\( r \) small constant, e.g., \( r = 4 \))
- \( \Gamma_n := \mathbb{U}(B_n) \)
- \( \Phi_n := \mathbb{U}(\text{hom}(B_n, B_r)) \)
- \( \Psi_n := \left[ \mathbf{v} \triangleleft \mathbb{U}(\mathbb{F}_3^r), \ \sigma \triangleleft S_r : \prod_{i=1}^{r} \chi_{\sigma(i)}^{v_i} \right] \quad (S_r: \text{\( r \)-permutations}) \)
  
  (unif. dist. over \( B_r \)-elements of Cayley-norm \( \leq r = : B_r \))

\[ B_n \xrightarrow{\approx} \varphi \triangleleft \text{hom}(B_n, B_r) \rightarrow B_r \]

\[ a \triangleleft \mathbb{U}(B_n) \xrightarrow{\varphi(a)\prod_{i=1}^{r} \chi_{\sigma(i)}^{v_i}} \quad (\mathbf{v} \triangleleft \mathbb{U}(\mathbb{F}_3^r), \ \sigma \triangleleft S_r) \]

\[ B_n \text{-LHN Assumption} \]

\[ \text{A}^{B_r}_{\varphi_{\text{PPT}}} \approx \mathbb{U}(B_n \times B_r), \]
$B_n$-LHN: Instantiating LHN over Burnside Groups

- $G_n := B_n, P_n := B_r$ ($r$ small constant, e.g., $r = 4$)
- $\Gamma_n := U(B_n)$
- $\Phi_n := U(\text{hom}(B_n, B_r))$
- $\Psi_n := \left[ v \leftarrow U(\mathbb{F}_3^r), \sigma \leftarrow S_r : \prod_{i=1}^r x_{\sigma(i)}^{v_i} \right] \quad (S_r: \text{r-permutations})$

$$\Psi_n := \left[ v \leftarrow U(\mathbb{F}_3^r), \sigma \leftarrow S_r : \prod_{i=1}^r x_{\sigma(i)}^{v_i} \right] \quad (S_r: \text{r-permutations})$$

(unif. dist. over $B_r$-elements of Cayley-norm $\leq r =: B_r$)

$$B_n \xrightarrow{\approx} \varphi \leftarrow \text{hom}(B_n, B_r) \rightarrow B_r$$

$$a \leftarrow U(B_n) \rightarrow \varphi(a)e, \quad (e \leftarrow B_r)$$

$B_n$-LHN Assumption

$A_{\varphi}^{B_r} \approx_{\text{PPT}} U(B_n \times B_r)$,

Nelly Fazio and Antonio R. Nicolosi
Cryptography via Burnside Groups
\( G_n := B_n, P_n := B_r \) (\( r \) small constant, e.g., \( r = 4 \))

\( \Gamma_n := \text{U}(B_n) \)

\( \Phi_n := \text{U}(\text{hom}(B_n, B_r)) \)

\( \Psi_n := \left[ v \leftarrow \text{U}(\mathbb{F}_3^r), \sigma \leftarrow S_r : \prod_{i=1}^r x_{\sigma(i)}^{v_i} \right] \quad (S_r: r\text{-permutations}) \)

(\text{unif. dist. over } B_r\text{-elements of Cayley-norm } \leq r =: B_r)
$B_n$-LHN Assumption

\[ \mathcal{A}_{\varphi}^{B_r} \approx_{\text{PPT}} \mathcal{U}(B_n \times B_r), \quad \text{any} \quad \varphi \in \text{Epi}(B_n, B_r) \]
1 Background
   - Burnside Groups ($B_n$)
   - Learning Burnside Homomorphisms with Noise ($B_n$-LHN)

2 Random Self-Reducibility of $B_n$-LHN

3 Cryptography via Burnside Groups
   - Minicrypt via Burnside Groups
   - Cryptomania via Burnside Groups? (future work)
Worst-case-to-average-case reduction for $B_n$-LHN: Solving random instances not easier than solving an arbitrary instance

Why does random self-reducibility matter?
- Hallmark of robust crypto assumptions (SIS, LWE, DLog, RSA)
- Desirable “all-or-nothing” hardness property: Either the problem is easy for (almost) all keys, or it is intractable for (almost) all keys
- Critical for actual cryptosystems: Generation of cryptographic keys amounts to sampling hard instances of underlying computational problem: by RSR ensures random instance suffices
In $B_n$-LHN, secret key is a $(B_n, B_r)$-homomorphism $\varphi$

$\Rightarrow$ Need to study $\text{hom}(B_n, B_r)$

Key fact: All Burnside groups are \textit{relatively free}

- For any group $P$ of exponent 3, any mapping of generators $x_1, \ldots, x_n$ into $P$ extends uniquely to a $(B_n, P)$-homomorphism
- So $|\text{hom}(B_n, P)| = 3^{|P|^n}$
- For $P = B_r$ ($r \ll n$), $|\text{hom}(B_n, B_r)| = 3^{\left(r + \binom{r}{2} + \binom{r}{3}\right)n}$

$\Rightarrow$ The key space in $B_n$-LHN is exponential in $n$ (security parameter)
Abelianization in $B_n$

- Abelianization of $B_n \equiv$ Quotient by its **commutator subgroup**:
  \[[B_n, B_n] \doteq \{ w_1^{-1} w_2^{-1} w_1 w_2 : w_1, w_2 \in B_n \}\]
  
  \[B_n/[B_n, B_n] \cong (\mathbb{F}_3^n, +)\]

- Abelianization map $\rho_n : B_n \to B_n/[B_n, B_n] \cong (\mathbb{F}_3^n, +)$
  \[\rho_n : \prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}} \mapsto (\alpha_1, \alpha_2, \ldots, \alpha_n)\]

- Abelianization of a $(B_n, B_r)$-homomorphism $\varphi$

  \[
  \begin{array}{ccc}
  B_n & \xrightarrow{\varphi} & B_r \\
  \downarrow \rho_n & & \downarrow \rho_r \\
  (\mathbb{F}_3^n, +) & \xrightarrow{\varphi} & (\mathbb{F}_3^r, +)
  \end{array}
  \]
Abelianization in $B_n$

- Abelianization of $B_n \equiv$ Quotient by its **commutator subgroup**:
  \[ [B_n, B_n] = \{ w_1^{-1} w_2^{-1} w_1 w_2 : w_1, w_2 \in B_n \} \]
  \[ B_n/[B_n, B_n] \cong (\mathbb{F}_3^n, +) \]

- Abelianization **map** $\rho_n : B_n \rightarrow B_n/[B_n, B_n] \cong (\mathbb{F}_3^n, +)$
  \[ \rho_n : \prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}} \mapsto (\alpha_1, \alpha_2, \ldots, \alpha_n) \]

- Abelianization of a $(B_n, B_r)$-homomorphism $\varphi$

\[ \begin{array}{ccc}
B_n & \xrightarrow{\varphi} & B_r \\
\downarrow{\rho_n} & & \downarrow{\rho_r} \\
(\mathbb{F}_3^n, +) & \xrightarrow{\varphi} & (\mathbb{F}_3^r, +) \end{array} \]
Abelianization in $B_n$

- Abelianization of $B_n \equiv$ Quotient by its **commutator subgroup**: 
  
  
  $[B_n, B_n] \doteq \{ w_1^{-1} w_2^{-1} w_1 w_2 : w_1, w_2 \in B_n \}$
  
  $B_n/[B_n, B_n] \cong (\mathbb{F}_3^n, +)$

- Abelianization **map** $\rho_n : B_n \to B_n/[B_n, B_n] \cong (\mathbb{F}_3^n, +)$

$$\rho_n : \prod_{i=1}^{n} x_i^{\alpha_i} \prod_{i<j} [x_i, x_j]^{\beta_{i,j}} \prod_{i<j<k} [x_i, x_j, x_k]^{\gamma_{i,j,k}} \mapsto (\alpha_1, \alpha_2, \ldots, \alpha_n)$$

- Abelianization of a $(B_n, B_r)$-**homomorphism** $\varphi$

\[
\begin{array}{ccc}
\rho_n & \rho_r \\
B_n & \varphi & B_r \\
(\mathbb{F}_3^n, +) & \overline{\varphi} & (\mathbb{F}_3^r, +)
\end{array}
\]
Q: Does abelianization reduce $B_n$-LHN to LWE over $\mathbb{F}_3$?

Recall: $a \xleftarrow{\$} U(B_n)$, $e = \prod_{i=1}^{r} x_{\sigma(i)}^{v_i}$, $(v_1, \ldots, v_r) \xleftarrow{\$} U(\mathbb{F}_3^r)$, $\sigma \xleftarrow{\$} S_r$
Q: Does abelianization reduce $B_n$-LHN to LWE over $\mathbb{F}_3$?

$$A_{\varphi}^{B_r} \left[ \text{i.e.,} (a, \varphi(a)e) \right] \approx_{\text{PPT}} U(B_n \times B_r)$$

Recall: $a \xleftarrow{\$} U(B_n), e = \prod_{i=1}^{r} x_{\sigma(i)}^{v_i} \quad \quad (v_1, \ldots, v_r) \xleftarrow{\$} U(\mathbb{F}_3^r), \quad \sigma \xleftarrow{\$} S_r$

Top row represents the $B_n$-LHN assumption
Abelianizing $B_n$-LHN vs. LWE with $p = 3$

- **Q:** Does abelianization reduce $B_n$-LHN to LWE over $\mathbb{F}_3$?

\[ \begin{align*}
A'^{B_r}_{\varphi} &= \begin{bmatrix} i.e., (a, \varphi(a)e) \end{bmatrix} \\
\approx_{\text{PPT}} &\mathcal{U}(B_n \times B_r) \\
\rho &\mathcal{U}(\mathbb{F}_3^n \times \mathbb{F}_3^r)
\end{align*} \]

- Recall: $a \overset{\$}{\leftarrow} \mathcal{U}(B_n)$, $e = \prod_{i=1}^{r} x_{\sigma(i)}^{v_i}$, $(v_1, \ldots, v_r) \overset{\$}{\leftarrow} \mathcal{U}(\mathbb{F}_3^r)$, $\sigma \overset{\$}{\leftarrow} S_r$

- Top row represents the $B_n$-LHN assumption
- Bottom row shows the result of abelianization
Abelianizing $B_n$-LHN vs. LWE with $ρ = 3$

- **Q:** Does abelianization reduce $B_n$-LHN to LWE over $\mathbb{F}_3$?

$$A_{ϕ}^{Br} \xrightarrow{[i.e., (a, ϕ(a)e) \]} U(B_n \times Br) \approx_{\text{PPT}} U(B_n \times Br)$$

- **Recall:** $a \xleftarrow{\$} U(B_n), e = \prod_{i=1}^{r} x_{σ(i)}^{v_i}$ $(v_1, \ldots, v_r) \xleftarrow{\$} U(\mathbb{F}_3^r), \ σ \xleftarrow{\$} S_r$
- Top row represents the $B_n$-LHN assumption
- Bottom row shows the result of abelianization
- Bottom distributions **identical**—cannot be distinguished!
  $$⇒$$ Abelianization does not help recognize $B_n$-LHN instances
Two main steps:

1. Start with a generic partial key-randomization trick

2. Show that this randomization is complete in the case of $B_n$-LHN with surjective secret key ($\varphi \in \text{Epi}(B_n, B_r)$)
Step 1: Domain Reshuffling

Lemma

Let $\alpha$ be a $G_n$-permutation, and $(a, b) \in G_n \times P_n$ be an LHN-instance sampled according to $A_{\varphi}^{\psi_n}$ ($b = \varphi(a)e$ for $e \xleftarrow{s} \Psi_n$). Let $a' = \alpha^{-1}(a)$. Then $(a', b) \in G_n \times P_n$ is sampled according to $A_{\varphi \circ \alpha}^{\psi_n}$.

Proof.

Observe that

\[
(a', b) = (a', \varphi(a) \cdot e) \\
= (a', \varphi \circ \alpha(\alpha^{-1}(a)) \cdot e) \\
= (a', \varphi \circ \alpha(a') \cdot e)
\]
Step 1: Domain Reshuffling

Lemma

Let $\alpha$ be a $G_n$-permutation, and $(a, b) \in G_n \times P_n$ be an LHN-instance sampled according to $\mathbf{A}_{\varphi}^{\psi_n}$ ($b = \varphi(a)e$ for $e \overset{\$}{\leftarrow} \psi_n$). Let $a' = \alpha^{-1}(a)$. Then $(a', b) \in G_n \times P_n$ is sampled according to $\mathbf{A}_{\varphi \circ \alpha}^{\psi_n}$.

Proof.

Observe that

$$(a', b) = (a', \varphi(a) \cdot e)$$

$$= (a', \varphi \circ \alpha(\alpha^{-1}(a)) \cdot e)$$

$$= (a', \varphi \circ \alpha(a') \cdot e)$$

\[\square\]
Domain Reshuffling provides some partial randomization for an instantiation of the abstract LHN problem.

For any $A_\varphi^n$, can transform an $A_\varphi^n$-instance into an $A_{\varphi \circ \alpha}^n$-instance, for any permutation $\alpha$.

In the case of $B_n$-LHN, this simple randomization is complete for the set of surjective homomorphisms:

**Lemma**

$$(\forall \varphi, \varphi' \in \text{Epi}(B_n, B_r))(\exists \alpha \in \text{Aut}(B_n))[\varphi' = \varphi \circ \alpha]$$
Proving Completeness

Claim

Given an arbitrary epimorphism $\varphi$ and a target epimorphism $\varphi^*$, there exist an automorphism $\alpha$ such that $\varphi^* = \varphi \circ \alpha$

Proof Idea

- Freeness of $B_n \Rightarrow \exists \beta \in \text{hom}(B_n, B_n)$ such that $\varphi^* = \varphi \circ \beta$

- Technical hurdle: $\beta$ need not be an automorphism!
- Solution: “Patch” $\beta$ into $\alpha \in \text{Aut}(B_n)$
“Patching argument” (omitted) hinges upon following technical lemma:

**Lemma**

Surjections $\varphi : B_n \to B_r$ are precisely the maps whose abelianization $\varphi'$ is also surjective

Proof ($\varphi \in \text{Epi}(B_n, B_r) \implies \varphi' \in \text{Epi}(\mathbb{F}_3^n, \mathbb{F}_3^r)$): Diagram chase
Proving Transitivity (cont’d)

Proof \( (\varphi' \in \text{Epi}(\mathbb{F}_3^n, \mathbb{F}_3^r) \implies \varphi \in \text{Epi}(B_n, B_r)) \)

- Let \( \{x_1, \ldots, x_n\} \) be \( B_n \) gener’s; define \( y_i = \varphi(x_i) \) and \( t_i = \rho_r(y_i) \)
- Thesis amounts to proving \( \{y_1, \ldots, y_n\} \) generates \( B_r \)
- By nilpotency of \( B_r \) (cf. next Lemma), suffices to show \( \{t_1, \ldots, t_n\} \) generates \( \mathbb{F}_3^r \)
- Diagram chase shows \( \rho_r \circ \varphi \) surj. \( \Rightarrow \) \( \{t_1, \ldots, t_n\} \) generates \( \mathbb{F}_3^r \)
Lemma

Let $G$ be a nilpotent group. If $\{y_1, \ldots, y_m\}$ generates $G$ modulo the commutator subgroup $[G, G]$, then $\{y_1, \ldots, y_m\}$ generates $G$.

Since $B_r$ has nilpotency class 3, and $B_r/[B_r, B_r] \cong F_3$, we get:

Corollary

Let $\rho_r : B_r \rightarrow F_3'$ denote abelianization, and $y_1, \ldots, y_m \in B_r$. Then $\{y_1, \ldots, y_m\}$ generates $B_r$ iff $\{\rho_r(y_1), \ldots, \rho_r(y_m)\}$ generates $F_3'$. 
Outline

1 Background
   - Burnside Groups ($B_n$)
   - Learning Burnside Homomorphisms with Noise ($B_n$-LHN)

2 Random Self-Reducibility of $B_n$-LHN

3 Cryptography via Burnside Groups
   - Minicrypt via Burnside Groups
   - Cryptomania via Burnside Groups? (future work)
Encryption

Fix an element $\tau \in B_r$ such that the shortest sequence of $x_i$ and $x_i^{-1}$ to express it is “large” (Cayley norm $\| \cdot \|_C$)

\[ t \in \{0, 1\} : \quad \text{Enc}_\varphi(t) = (a, \tau b) \quad (a, b) \leftarrow A_n^{B_r} \]

Decryption

\[ \text{Dec}_\varphi(a, b') = \begin{cases} 0 & \text{if } \|\varphi(a), b'\|_C \text{ “small”} \\ 1 & \text{o/w} \end{cases} \]
Encryption

Fix an element $\tau \in B_r$ such that the shortest sequence of $x_i$ and $x_i^{-1}$ to express it is “large” (Cayley norm $\| \cdot \|_C$)

$$t \in \{0, 1\} : \quad \text{Enc}_\varphi(t) = (a, \tau b) \quad (a, b) \leftarrow A_n^{B_r}$$

Decryption

$$\text{Dec}_\varphi(a, b') = \begin{cases} 
0 & \text{if } \| \varphi(a), b' \|_C \text{ “small”} \\
1 & \text{o/w}
\end{cases}$$
Summary

- Algebraic generalization of the LWE problem to an abstract group-theoretic setting
- Exploration of the cryptographic viability of Burnside groups
  - Technical lemmas about homomorphisms between Burnside groups of exponent three
- Evidence to the hardness of the $B_n$-LHN problem of
  - Random Self-Reducibility: Solving random instances is as hard as solving arbitrary ones
Group operation in $B_n$: Example

\[ x_1^{-1} x_3 [x_2, x_3] \cdot x_1 x_2 [x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1 [x_2, x_3] [x_2, x_3, x_1] x_2 [x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1 [x_2, x_3] [x_1, x_2, x_3] x_2 [x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3] [x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3]^{-1} = \]
\[ x_1^{-1} x_3 x_1 x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_1^{-1} x_1 x_3 [x_3, x_1] x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_3 [x_1, x_3]^{-1} x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2 [x_1, x_3]^{-1} [x_1, x_3, x_2]^{-1} [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2 [x_1, x_3]^{-1} [x_1, x_2, x_3] [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2 [x_1, x_3]^{-1} [x_2, x_3] [x_1, x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_2 x_3 [x_3, x_2] [x_1, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3 [x_2, x_3]^{-1} [x_1, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3 [x_1, x_3]^{-1} [x_2, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3 [x_1, x_3]^{-1} \]

Nelly Fazio and Antonio R. Nicolosi

Cryptography via Burnside Groups
Group operation in $B_n$: Example

\[ \begin{align*}
    x_1^{-1} x_3 [x_2, x_3] & \cdot x_1 x_2 [x_1, x_2, x_3] = \\
    x_1^{-1} x_3 x_1 [x_2, x_3][x_2, x_3, x_1] x_2 [x_1, x_2, x_3] &= \\
    x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3] & = \\
    x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3]^{-1} & = \\
    x_1^{-1} x_3 x_1 x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} &= \\
    x_1^{-1} x_1 x_3 [x_3, x_1] x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} &= \\
    x_3 [x_1, x_3]^{-1} x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} &= \\
    x_3 x_2 [x_1, x_3]^{-1} [x_1, x_3, x_2]^{-1} [x_2, x_3] [x_1, x_2, x_3]^{-1} &= \\
    x_3 x_2 [x_1, x_3]^{-1} [x_1, x_2, x_3] [x_2, x_3] [x_1, x_2, x_3]^{-1} &= \\
    x_3 x_2 [x_1, x_3]^{-1} [x_2, x_3] [x_1, x_2, x_3]^{-1} &= \\
    x_2 x_3 [x_3, x_2] [x_1, x_3]^{-1} [x_2, x_3] &= \\
    x_2 x_3 [x_2, x_3]^{-1} [x_1, x_3]^{-1} [x_2, x_3] &= \\
    x_2 x_3 [x_1, x_3]^{-1} [x_2, x_3]^{-1} [x_2, x_3] &= \\
    x_2 x_3 [x_1, x_3]^{-1}
\end{align*} \]
Group operation in $B_n$: Example

$$x_1^{-1} x_3 [x_2, x_3] \cdot x_1 x_2 [x_1, x_2, x_3] =$$

$$x_1^{-1} x_3 x_1 [x_2, x_3] [x_2, x_3, x_1] x_2 [x_1, x_2, x_3] =$$

$$x_1^{-1} x_3 x_1 [x_2, x_3] [x_1, x_2, x_3] x_2 [x_1, x_2, x_3] =$$

$$x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3] [x_1, x_2, x_3] =$$

$$x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3]^{-1} =$$

$$x_1^{-1} x_3 x_1 x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} =$$

$$x_1^{-1} x_1 x_3 [x_3, x_1] x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} =$$

$$x_3 [x_1, x_3]^{-1} x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} =$$

$$x_3 x_2 [x_1, x_3]^{-1} [x_1, x_3, x_2]^{-1} [x_2, x_3] [x_1, x_2, x_3]^{-1} =$$

$$x_3 x_2 [x_1, x_3]^{-1} [x_1, x_2, x_3] [x_2, x_3] [x_1, x_2, x_3]^{-1} =$$

$$x_3 x_2 [x_1, x_3]^{-1} [x_2, x_3] [x_1, x_2, x_3]^{-1} =$$

$$x_2 x_3 [x_3, x_2] [x_1, x_3]^{-1} [x_2, x_3] =$$

$$x_2 x_3 [x_2, x_3]^{-1} [x_1, x_3]^{-1} [x_2, x_3] =$$

$$x_2 x_3 [x_1, x_3]^{-1} [x_2, x_3]^{-1} [x_2, x_3] =$$

$$x_2 x_3 [x_1, x_3]^{-1}$$
Group operation in $B_n$: Example

$$x_1^{-1}x_3[x_2, x_3] \cdot x_1x_2[x_1, x_2, x_3] =$$

$$x_1^{-1}x_3x_1[x_2, x_3][x_2, x_3, x_1]x_2[x_1, x_2, x_3] =$$

$$x_1^{-1}x_3x_1[x_2, x_3][x_1, x_2, x_3]x_2[x_1, x_2, x_3] =$$

$$x_1^{-1}x_3x_1[x_2, x_3]x_2[x_1, x_2, x_3][x_1, x_2, x_3] =$$

$$x_1^{-1}x_3x_1[x_2, x_3]x_2[x_1, x_2, x_3][x_1, x_2, x_3]^{-1} =$$

$$x_1^{-1}x_3x_1x_2[x_2, x_3][x_1, x_2, x_3]^{-1} =$$

$$x_1^{-1}x_3[x_3, x_1]x_2[x_2, x_3][x_1, x_2, x_3]^{-1} =$$

$$x_3[x_1, x_3]^{-1}x_2[x_2, x_3][x_1, x_2, x_3]^{-1} =$$

$$x_3x_2[x_1, x_3]^{-1}[x_1, x_3, x_2]^{-1}[x_2, x_3][x_1, x_2, x_3]^{-1} =$$

$$x_3x_2[x_1, x_3]^{-1}[x_1, x_2, x_3][x_2, x_3][x_1, x_2, x_3]^{-1} =$$

$$x_3x_2[x_1, x_3]^{-1}[x_2, x_3][x_1, x_2, x_3][x_1, x_2, x_3]^{-1} =$$

$$x_2x_3[x_1, x_3]^{-1}[x_2, x_3] =$$

$$x_2x_3[x_2, x_3]^{-1}[x_1, x_3]^{-1}[x_2, x_3] =$$

$$x_2x_3[x_1, x_3]^{-1}[x_2, x_3]^{-1}[x_2, x_3] =$$

$$x_2x_3[x_1, x_3]^{-1}$$
Group operation in $B_n$: Example

\[ x_1^{-1} x_3[x_2, x_3] \cdot x_1 x_2[x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1[x_2, x_3][x_2, x_3, x_1] x_2[x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1[x_2, x_3][x_1, x_2, x_3] x_2[x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1[x_2, x_3] x_2[x_1, x_2, x_3][x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1[x_2, x_3] x_2[x_1, x_2, x_3]^{-1} = \]
\[ x_1^{-1} x_3 x_1 x_2[x_2, x_3][x_1, x_2, x_3]^{-1} = \]
\[ x_1^{-1} x_1 x_3[x_3, x_1] x_2[x_2, x_3][x_1, x_2, x_3]^{-1} = \]
\[ x_3[x_1, x_3]^{-1} x_2[x_2, x_3][x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2[x_1, x_3]^{-1} [x_1, x_3, x_2]^{-1} [x_2, x_3][x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2[x_1, x_3]^{-1} [x_1, x_2, x_3][x_2, x_3][x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2[x_1, x_3]^{-1} [x_2, x_3][x_1, x_2, x_3][x_1, x_2, x_3]^{-1} = \]
\[ x_2 x_3[x_3, x_2] [x_1, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3[x_2, x_3]^{-1} [x_1, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3[x_1, x_3]^{-1} [x_2, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3[x_1, x_3]^{-1} \]
Group operation in $B_n$: Example

\[ x_1^{-1} x_3 [x_2, x_3] \cdot x_1 x_2 [x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1 [x_2, x_3] [x_2, x_3, x_1] x_2 [x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1 [x_2, x_3] [x_1, x_2, x_3] x_2 [x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3] [x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3]^{-1} = \]
\[ x_1^{-1} x_3 x_1 x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_1^{-1} x_1 x_3 [x_3, x_1] x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_3 [x_1, x_3]^{-1} x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2 [x_1, x_3]^{-1} [x_1, x_3, x_2]^{-1} [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2 [x_1, x_3]^{-1} [x_1, x_2, x_3] [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2 [x_1, x_3]^{-1} [x_1, x_2, x_3] [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_2 x_3 [x_3, x_2] [x_1, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3 [x_2, x_3]^{-1} [x_1, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3 [x_1, x_3]^{-1} x_2 [x_2, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3 [x_1, x_3]^{-1} \]
Group operation in $B_n$: Example

\[
x_1^{-1}x_3[x_2,x_3] \cdot x_1x_2[x_1,x_2,x_3] =
\]
\[
x_1^{-1}x_3x_1[x_2,x_3][x_2,x_3,x_1]x_2[x_1,x_2,x_3] =
\]
\[
x_1^{-1}x_3x_1[x_2,x_3][x_1,x_2,x_3]x_2[x_1,x_2,x_3] =
\]
\[
x_1^{-1}x_3x_1[x_2,x_3]x_2[x_1,x_2,x_3][x_1,x_2,x_3] =
\]
\[
x_1^{-1}x_3x_1[x_2,x_3]x_2[x_1,x_2,x_3]^{-1} =
\]
\[
x_1^{-1}x_3x_1x_2[x_2,x_3][x_1,x_2,x_3]^{-1} =
\]
\[
x_1^{-1}x_1x_3[x_3,x_1]x_2[x_2,x_3][x_1,x_2,x_3]^{-1} =
\]
\[
x_3[x_1,x_3]^{-1}x_2[x_2,x_3][x_1,x_2,x_3]^{-1} =
\]
\[
x_3x_2[x_1,x_3]^{-1}[x_1,x_3,x_2]^{-1}[x_2,x_3][x_1,x_2,x_3]^{-1} =
\]
\[
x_3x_2[x_1,x_3]^{-1}[x_1,x_2,x_3][x_2,x_3][x_1,x_2,x_3]^{-1} =
\]
\[
x_3x_2[x_1,x_3]^{-1}[x_2,x_3][x_1,x_2,x_3][x_1,x_2,x_3]^{-1} =
\]
\[
x_2x_3[x_3,x_2][x_1,x_3]^{-1}[x_2,x_3] =
\]
\[
x_2x_3[x_2,x_3]^{-1}[x_1,x_3]^{-1}[x_2,x_3] =
\]
\[
x_2x_3[x_1,x_3]^{-1}[x_2,x_3]^{-1}[x_2,x_3] =
\]
\[
x_2x_3[x_1,x_3]^{-1}
\]
Group operation in $B_n$: Example

\[
x_1^{-1}x_3[x_2, x_3] \cdot x_1x_2[x_1, x_2, x_3] = \]

\[
x_1^{-1}x_3x_1[x_2, x_3][x_2, x_3, x_1]x_2[x_1, x_2, x_3] = \]

\[
x_1^{-1}x_3x_1[x_2, x_3][x_1, x_2, x_3]x_2[x_1, x_2, x_3] = \]

\[
x_1^{-1}x_3x_1[x_2, x_3]x_2[x_1, x_2, x_3][x_1, x_2, x_3] = \]

\[
x_1^{-1}x_3x_1[x_2, x_3]x_2[x_1, x_2, x_3]^{-1} = \]

\[
x_1^{-1}x_3x_1x_2[x_2, x_3][x_1, x_2, x_3]^{-1} = \]

\[
x_1^{-1}x_1x_3[x_3, x_1]x_2[x_2, x_3][x_1, x_2, x_3]^{-1} = \]

\[
x_3[x_1, x_3]^{-1}x_2[x_2, x_3][x_1, x_2, x_3]^{-1} = \]

\[
x_3x_2[x_1, x_3]^{-1}[x_1, x_3, x_2]^{-1}[x_2, x_3][x_1, x_2, x_3]^{-1} = \]

\[
x_3x_2[x_1, x_3]^{-1}[x_1, x_2, x_3][x_2, x_3][x_1, x_2, x_3]^{-1} = \]

\[
x_3x_2[x_1, x_3]^{-1}[x_2, x_3][x_1, x_2, x_3][x_1, x_2, x_3]^{-1} = \]

\[
x_2x_3[x_3, x_2][x_1, x_3]^{-1}[x_2, x_3] = \]

\[
x_2x_3[x_2, x_3]^{-1}[x_1, x_3]^{-1}[x_2, x_3] = \]

\[
x_2x_3[x_1, x_3]^{-1}[x_2, x_3]^{-1}[x_2, x_3] = \]

\[
x_2x_3[x_1, x_3]^{-1} \]
Group operation in $B_n$: Example

\[
\begin{align*}
  x_1^{-1} x_3 [x_2, x_3] & \quad \cdot \quad x_1 x_2 [x_1, x_2, x_3] = \\
  x_1^{-1} x_3 x_1 [x_2, x_3] [x_2, x_3, x_1] x_2 [x_1, x_2, x_3] = \\
  x_1^{-1} x_3 x_1 [x_2, x_3] [x_1, x_2, x_3] x_2 [x_1, x_2, x_3] = \\
  x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3] [x_1, x_2, x_3] = \\
  x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3]^{-1} = \\
  x_1^{-1} x_3 x_1 x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
  x_1^{-1} x_1 x_3 [x_3, x_1] x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
  x_3 [x_1, x_3]^{-1} x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
  x_3 x_2 [x_1, x_3]^{-1} [x_1, x_3, x_2]^{-1} [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
  x_3 x_2 [x_1, x_3]^{-1} [x_1, x_2, x_3] [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
  x_3 x_2 [x_1, x_3]^{-1} [x_2, x_3] [x_1, x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
  x_2 x_3 [x_3, x_2] [x_1, x_3]^{-1} [x_2, x_3] = \\
  x_2 x_3 [x_2, x_3]^{-1} [x_1, x_3]^{-1} [x_2, x_3] = \\
  x_2 x_3 [x_1, x_3]^{-1} [x_2, x_3]^{-1} [x_2, x_3] = \\
  x_2 x_3 [x_1, x_3]^{-1}
\end{align*}
\]
Group operation in $B_n$: Example

\[
\begin{align*}
&x_1^{-1}x_3 [x_2, x_3] \cdot x_1 x_2 [x_1, x_2, x_3] = \\
&x_1^{-1}x_3 x_1 [x_2, x_3] [x_2, x_3, x_1] x_2 [x_1, x_2, x_3] = \\
&x_1^{-1}x_3 x_1 [x_2, x_3] [x_1, x_2, x_3] x_2 [x_1, x_2, x_3] = \\
&x_1^{-1}x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3] [x_1, x_2, x_3] = \\
&x_1^{-1}x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3]^{-1} = \\
&x_1^{-1}x_3 x_1 x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
&x_1^{-1}x_1 x_3 [x_3, x_1] x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
&x_3 [x_1, x_3]^{-1} x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
&x_3 x_2 [x_1, x_3]^{-1} [x_1, x_3, x_2]^{-1} [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
&x_3 x_2 [x_1, x_3]^{-1} [x_1, x_2, x_3] [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
&x_3 x_2 [x_1, x_3]^{-1} [x_2, x_3] [x_1, x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
&x_2 x_3 [x_3, x_2] [x_1, x_3]^{-1} [x_2, x_3] = \\
&x_2 x_3 [x_2, x_3]^{-1} [x_1, x_3]^{-1} [x_2, x_3] = \\
&x_2 x_3 [x_1, x_3]^{-1} [x_2, x_3]^{-1} [x_2, x_3] = \\
&x_2 x_3 [x_1, x_3]^{-1}
\end{align*}
\]
Group operation in $B_n$: Example

\[
x_1^{-1} x_3 [x_2, x_3] \cdot x_1 x_2 [x_1, x_2, x_3] = \\
x_1^{-1} x_3 x_1 [x_2, x_3] [x_2, x_3, x_1] x_2 [x_1, x_2, x_3] = \\
x_1^{-1} x_3 x_1 [x_2, x_3] [x_1, x_2, x_3] x_2 [x_1, x_2, x_3] = \\
x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3] [x_1, x_2, x_3] = \\
x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3]^{-1} = \\
x_1^{-1} x_3 x_1 x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
x_1^{-1} x_3 x_1 x_3 [x_3, x_1] x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
x_3 [x_1, x_3]^{-1} x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
x_3 x_2 [x_1, x_3]^{-1} [x_1, x_3, x_2]^{-1} [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
x_3 x_2 [x_1, x_3]^{-1} [x_1, x_2, x_3] [x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
x_3 x_2 [x_1, x_3]^{-1} [x_2, x_3] [x_1, x_2, x_3] [x_1, x_2, x_3]^{-1} = \\
x_2 x_3 [x_1, x_3]^{-1} [x_2, x_3] [x_1, x_3]^{-1} [x_2, x_3] = \\
x_2 x_3 [x_1, x_3]^{-1} [x_2, x_3] [x_1, x_3]^{-1} [x_2, x_3] = \\
x_2 x_3 [x_1, x_3]^{-1} [x_2, x_3]^{-1} [x_2, x_3] = \\
x_2 x_3 [x_1, x_3]^{-1}
\]
Group operation in $B_n$: Example

$$x_1^{-1}x_3[x_2, x_3] \cdot x_1x_2[x_1, x_2, x_3] =$$

$$x_1^{-1}x_3x_1[x_2, x_3][x_2, x_3, x_1]x_2[x_1, x_2, x_3] =$$

$$x_1^{-1}x_3x_1[x_2, x_3][x_1, x_2, x_3]x_2[x_1, x_2, x_3] =$$

$$x_1^{-1}x_3x_1[x_2, x_3]x_2[x_1, x_2, x_3][x_1, x_2, x_3] =$$

$$x_1^{-1}x_3x_1[x_2, x_3]x_2[x_1, x_2, x_3]^{-1} =$$

$$x_1^{-1}x_3x_1x_2[x_2, x_3][x_1, x_2, x_3]^{-1} =$$

$$x_1^{-1}x_3x_1[x_3, x_1]x_2[x_2, x_3][x_1, x_2, x_3]^{-1} =$$

$$x_3[x_1, x_3]^{-1}x_2[x_2, x_3][x_1, x_2, x_3]^{-1} =$$

$$x_3x_2[x_1, x_3]^{-1}[x_1, x_3, x_2]^{-1}[x_2, x_3][x_1, x_2, x_3]^{-1} =$$

$$x_3x_2[x_1, x_3]^{-1}[x_1, x_2, x_3][x_2, x_3][x_1, x_2, x_3]^{-1} =$$

$$x_3x_2[x_1, x_3]^{-1}[x_2, x_3][x_1, x_2, x_3][x_1, x_2, x_3]^{-1} =$$

$$x_2x_3[x_3, x_2][x_1, x_3]^{-1}[x_2, x_3] =$$

$$x_2x_3[x_2, x_3]^{-1}[x_1, x_3]^{-1}[x_2, x_3] =$$

$$x_2x_3[x_1, x_3]^{-1}[x_2, x_3]^{-1}[x_2, x_3] =$$

$$x_2x_3[x_1, x_3]^{-1}$$
Group operation in $B_n$: Example

\[ x_1^{-1} x_3[x_2, x_3] \cdot x_1 x_2[x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1[x_2, x_3][x_2, x_3, x_1] x_2[x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1[x_2, x_3][x_1, x_2, x_3] x_2[x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1[x_2, x_3] x_2[x_1, x_2, x_3][x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1[x_2, x_3] x_2[x_1, x_2, x_3]^{-1} = \]
\[ x_1^{-1} x_3 x_1 x_2[x_2, x_3][x_1, x_2, x_3]^{-1} = \]
\[ x_1^{-1} x_1 x_3[x_3, x_1] x_2[x_2, x_3][x_1, x_2, x_3]^{-1} = \]
\[ x_3[x_1, x_3]^{-1} x_2[x_2, x_3][x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2[x_1, x_3]^{-1} [x_1, x_3, x_2]^{-1} [x_2, x_3][x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2[x_1, x_3]^{-1} [x_1, x_2, x_3][x_2, x_3][x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2[x_1, x_3]^{-1} [x_2, x_3][x_1, x_2, x_3][x_1, x_2, x_3]^{-1} = \]
\[ x_2 x_3[x_3, x_2][x_1, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3[x_2, x_3]^{-1} [x_1, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3[x_1, x_3]^{-1} [x_2, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3[x_1, x_3]^{-1} \]
Group operation in $B_n$: Example

\[
x_1^{-1} x_3 [x_2, x_3] \cdot x_1 x_2 [x_1, x_2, x_3] =
\]
\[
x_1^{-1} x_3 x_1 [x_2, x_3] [x_2, x_3, x_1] x_2 [x_1, x_2, x_3] =
\]
\[
x_1^{-1} x_3 x_1 [x_2, x_3] [x_1, x_2, x_3] x_2 [x_1, x_2, x_3] =
\]
\[
x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3] [x_1, x_2, x_3] =
\]
\[
x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3]^{-1} =
\]
\[
x_1^{-1} x_3 x_1 x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} =
\]
\[
x_1^{-1} x_1 x_3 [x_3, x_1] x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} =
\]
\[
x_3 [x_1, x_3]^{-1} x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} =
\]
\[
x_3 x_2 [x_1, x_3]^{-1} [x_1, x_3, x_2]^{-1} [x_2, x_3] [x_1, x_2, x_3]^{-1} =
\]
\[
x_3 x_2 [x_1, x_3]^{-1} [x_1, x_2, x_3] [x_2, x_3] [x_1, x_2, x_3]^{-1} =
\]
\[
x_3 x_2 [x_1, x_3]^{-1} [x_2, x_3] [x_1, x_2, x_3] [x_1, x_2, x_3]^{-1} =
\]
\[
x_2 x_3 [x_3, x_2] [x_1, x_3]^{-1} [x_2, x_3] =
\]
\[
x_2 x_3 [x_2, x_3]^{-1} [x_1, x_3]^{-1} [x_2, x_3] =
\]
\[
x_2 x_3 [x_1, x_3]^{-1} [x_2, x_3]^{-1} [x_2, x_3] =
\]
\[
x_2 x_3 [x_1, x_3]^{-1} [x_2, x_3]^{-1} [x_2, x_3] =
\]
Group operation in $B_n$: Example

\[ x_1^{-1} x_3 [x_2, x_3] \cdot x_1 x_2 [x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1 [x_2, x_3] [x_2, x_3, x_1] x_2 [x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1 [x_2, x_3] [x_1, x_2, x_3] x_2 [x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3] [x_1, x_2, x_3] = \]
\[ x_1^{-1} x_3 x_1 [x_2, x_3] x_2 [x_1, x_2, x_3]^{-1} = \]
\[ x_1^{-1} x_3 x_1 x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_1^{-1} x_3 x_1 [x_3, x_1] x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_3 [x_1, x_3]^{-1} x_2 [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2 [x_1, x_3]^{-1} [x_1, x_3, x_2]^{-1} [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2 [x_1, x_3]^{-1} [x_1, x_2, x_3] [x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_3 x_2 [x_1, x_3]^{-1} [x_2, x_3] [x_1, x_2, x_3] [x_1, x_2, x_3]^{-1} = \]
\[ x_2 x_3 [x_3, x_2] [x_1, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3 [x_2, x_3]^{-1} [x_1, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3 [x_1, x_3]^{-1} [x_2, x_3]^{-1} [x_2, x_3] = \]
\[ x_2 x_3 [x_1, x_3]^{-1} \]