Generalized Feistel Networks with Optimal Diffusion

Léo Perrin

DTU, Lyngby
Inria, Paris

Dagstuhl 2018 (seminar-18021)
In this talk

- A new type of generalized Feistel Networks
- Linear layer design
- Wide block cipher/sponge permutation blueprint
- Fibonacci numbers!
<table>
<thead>
<tr>
<th></th>
<th>Introduction</th>
<th>Observations on GFNs</th>
<th>Multi-Rotating Feistel Network (MRFN)</th>
<th>Possible Applications</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Observations on GFNs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Multi-Rotating Feistel Network (MRFN)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Possible Applications</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Conclusion</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
First GFN

Source: Generalized Feistel networks, K. Nyberg (1996)
Basic GFN

Improved GFN

How long does it take for each input word to influence each output word?

*The state consists of $2^b$ branches.*
Diffusion in Generalized Feistel networks

How long does it take for each input word to influence each output word?

*The state consists of $2b$ branches.*

Nyberg/Type-II GFN:

$\approx 2b$ rounds
Diffusion in Generalized Feistel networks

How long does it take for each input word to influence each output word?

*The state consists of $2^b$ branches.*

Nyberg/Type-II GFN:  
$\approx 2^b$ rounds

TWINE-like GFN: $\approx 2\log_2(b)$ rounds
Optimal Diffusion

The best we can achieve is for $X_0^i$ to influence $\phi_{i+2}$ branches at round $i$, where

$$\phi_0 = 0, \quad \phi_1 = 1, \quad \phi_{i+2} = \phi_{i+1} + \phi_i.$$
## Diffusion in GFNs

<table>
<thead>
<tr>
<th>$b$</th>
<th>8</th>
<th>16</th>
<th>32</th>
<th>64</th>
<th>128</th>
<th>..</th>
<th>2048</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nyberg Type-II/Nyberg</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>128</td>
<td>256</td>
<td>4096</td>
<td></td>
</tr>
<tr>
<td>TWINE-like</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>Optimal</td>
<td>6</td>
<td>8</td>
<td>9</td>
<td>11</td>
<td>12</td>
<td>18</td>
<td></td>
</tr>
</tbody>
</table>

Number of rounds for full diffusion.
Can we reach the Fibonacci-based bound?

Can we have an easy to implement $\pi$?
Can we reach the Fibonacci-based bound?

Can we have an easy to implement $\pi$?

Yes (for both)
Outline

1 Introduction

2 Observations on GFNs

3 Multi-Rotating Feistel Network (MRFN)

4 Possible Applications

5 Conclusion
General Structure

- Number of branches: $2b$
- Number of rounds: $r$
- $w$-bit permutations $f^i_j$ ($i < r$, $j < b$)
- Sequence $s^i$ of rotations of $b$ words.

The round $i$ of a MRFN with $b = 4$ and $s^i = 1$ is:
Some Observations

- Both a Feistel network and a GFN
- $\pi$ is very simple (1 word-wise rotation per round)
- Round function depends on the round index.
- Interesting case: $s^i = \phi_i$. 
Some Observations

- Both a Feistel network and a GFN
- \( \pi \) is very simple (1 word-wise rotation per round)
- Round function depends on the round index.
- Interesting case: \( s^i = \phi_i \).

Fibonacci Case

A MRFN with \( s^i = \phi_i \) has optimal diffusion.
At round 0, $X^0_0$ has touched the first $\phi_1 = 1$ branches of one side.
Example with 12 branches

\[ \phi_0 = 0 \]
\[ \phi_1 = 1 \]
\[ \phi_2 = 1 \]
\[ \phi_3 = 2 \]
\[ \phi_4 = 3 \]
Implementation

VRound function operating on $2b \cdot w$ bit internal state.
Implementation

VRound function operating on $2w$ bit internal state.

1. copy
2. parallel layer of $f_i$ 3. rotations
4. XOR
5. swap
6. finished!

$b$

$w$
Implementation

VRound function operating on $2^w$ bit internal state.

1. copy
2. parallel layer of $f^i$
3. rotations
4. XOR
5. swap
6. finished!

2. parallel layer of $f^i$
Implementation

VRound function operating on $2^w$ bit internal state.

1. copy
2. parallel layer of $f_i$
3. rotations
4. XOR
5. swap
6. finished!

3. rotations
Implementation

4. XOR
5. swap
Implementation

6. finished!
Some Observations

- $s^i$ and $s^i + (-\ell)^i \mod b$ are equivalent

- if $\gcd(s^i, b) \neq 1$ for all $i$, no full diffusion!

- Importance of the choice of $\{s^i\}_{i\geq 0}$
Security

- If \( s^i = \phi_i \), then full diffusion in \( \approx \Lambda(n) \) rounds, where \( \Lambda(x) = i \) if \( \phi_{i-1} < x \leq \phi_i \) (optimal).

- If \( s^{2i} = 0 \) and \( i_{2i+1} = 2^i \), then full diffusion in \( \approx 2 \log_2(n) \) rounds (like TWINE).

- Both are quickly safe from miss-in-the-middle based impossible differential attacks and MitM!
Security

- If $s^i = \phi_i$, then full diffusion in $\approx \Lambda(n)$ rounds, where $\Lambda(x) = i$ if $\phi_{i-1} < x \leq \phi_i$ (optimal).

- If $s^{2i} = 0$ and $i_{2i+1} = 2^i$, then full diffusion in $\approx 2 \log_2(n)$ rounds (like TWINE).

- Both are quickly safe from miss-in-the-middle based impossible differential attacks and MitM!

- When $s^i = \phi_i$, bad truncated differential with 2 active S-Boxes/round.
Security

- If $s^i = \phi_i$, then full diffusion in $\approx \Lambda(n)$ rounds, where $\Lambda(x) = i$ if $\phi_{i-1} < x \leq \phi_i$ (optimal).

- If $s^{2i} = 0$ and $i_{2i+1} = 2^i$, then full diffusion in $\approx 2 \log_2(n)$ rounds (like TWINE).

- Both are quickly safe from miss-in-the-middle based impossible differential attacks and MitM!

- When $s^i = \phi_i$, bad truncated differential with 2 active S-Boxes/round.

Open Problem 1
Differential/Linear bound?

Open Problem 2
Choice of $\{s^i\}_{i \geq 0}$?
Outline

1. Introduction
2. Observations on GFNs
3. Multi-Rotating Feistel Network (MRFN)
4. Possible Applications
5. Conclusion
GFN-based Linear Layers

- Use linear $\{f^i\}_{i \geq 0}; s^i = \phi_i$
- $n$-bit block divided into $2b$ branches of $w$ bits uses:

$$\frac{w^2}{2} \times b \times 2 \log_2(b) \text{ XORs}.$$
GFN-based Linear Layers

- Use linear $\{f^i\}_{i \geq 0}$; $s^i = \phi_i$
- $n$-bit block divided into $2b$ branches of $w$ bits uses:
  $$\left(\frac{w^2}{2}\right) \times b \times 2\log_2(b) \text{ XORs}.$$  
  
  If we fix $w$ to a small value, then the number of XORs scales with $n \log_2(n)$ rather than $n^2$.  

GFN-based Linear Layers

- Use linear \( \{ f^i \} \}_{i \geq 0}; s^i = \phi_i \)
- \( n \)-bit block divided into \( 2b \) branches of \( w \) bits uses:

\[
\frac{w^2}{2} \times b \times 2^{\log_2(b)} \text{ XORs}.
\]

- If we fix \( w \) to a small value, then the number of XORs scales with \( n \log_2(n) \) rather than \( n^2 \).
- Practical gains even for \( n = 256 \):

  *Improvements to the Linear Layer of LowMC: A Faster Picnic*, with Angela Promitzer, Sebastian Ramacher and Christian Rechberger (2017/448)
Example of Linear Layer

- $n = 256$
- $w = 4$
- $b = 32$

$i = 0$
Example of Linear Layer

- \( n = 256 \)
- \( w = 4 \)
- \( b = 32 \)

\( i = 1 \)
Example of Linear Layer

- $n = 256$
- $w = 4$
- $b = 32$

- $i = 2$
Example of Linear Layer

- \( n = 256 \)
- \( w = 4 \)
- \( b = 32 \)

\( i = 3 \)
Example of Linear Layer

- $n = 256$
- $w = 4$
- $b = 32$

$i = 4$
Example of Linear Layer

- $n = 256$
- $w = 4$
- $b = 32$

$i = 5$
Example of Linear Layer

- $n = 256$
- $w = 4$
- $b = 32$

$i = 6$
Example of Linear Layer

- $n = 256$
- $w = 4$
- $b = 32$

$i = 7$
Example of Linear Layer

- $n = 256$
- $w = 4$
- $b = 32$

$i = 8$
Example of Linear Layer

- \( n = 256 \)
- \( w = 4 \)
- \( b = 32 \)

\( i = 9 \)
Example of Linear Layer

- $n = 256$
- $w = 4$
- $b = 32$
- $i = 10$
Sponge function?

- $n = 384$, with $b = 64$ and $w = 3$
- $f^i_j(x) = \chi_3(x \oplus c^i_j)$
- $s^{2i} = 0$, $s^{2i+1} = 2^i$ for $0 \leq i < 2 \log_2(b) = 12$, then repeat (4? times):
  
  
  $$s = \{0, 1, 0, 2, 0, 4, 0, 8, 0, 16, 0, 32\}$$
Sponge function?

- $n = 384$, with $b = 64$ and $w = 3$
- $f^i_j(x) = \chi_3(x \oplus c^i_j)$
- $s^{2i} = 0, s^{2i+1} = 2^i$ for $0 \leq i < 2\log_2(b) = 12$, then repeat (4? times):
  $$s = \{0, 1, 0, 2, 0, 4, 0, 8, 0, 16, 0, 32\}$$

Efficiency estimates

On 64-bit processors, for each round:

- 3 word copies
- 3 word-wise AND
- 3+3+3 word-wise XORs
- **Maybe** safe for **48 rounds** if $\geq 8$ active $f$ functions/round on average.
MiMC-like construction where $f^i_j(x) = (x + c^i_j)^3$ (what Arnab just presented).
MiMC-like construction where $f^i_j(x) = (x + c^i_j)^3$ (what Arnab just presented).

- You tell me!
Fun stuff happens when we allow the use of different permutations in each round!

Open problems

1. What are good sequences of rotations?
2. How to bound number of active $f$ functions?
3. What can we use it for?
4. What happens in other structures (SPN? ARX?) when the linear layers are round-dependent?
Fun stuff happens when we allow the use of different permutations in each round!

Open problems

1. What are good sequences of rotations?

2. How to bound number of active $f$ functions?

3. What can we use it for?

4. What happens in other structures (SPN? ARX?) when the linear layers are round-dependent?

Thank you!