

Graduation Thesis

Run length limited de Bruijn sequences for quantum communication

Nguyen Tien Long

Supervisors: Dr. Vu Van Khu

Dr. Tran Vinh Duc

School of Information and Communication Technology.

August 18, 2022

Table of Contents

- 1 Introduction
- 2 Run length limited de Bruijn sequences
- 3 Rate and Maximal Asymptotic Rate
- 4 Encoding and Decoding Algorithm
- 5 Conclusion

Table of Contents

- 1 Introduction
- 2 Run length limited de Bruijn sequences
- 3 Rate and Maximal Asymptotic Rate
- 4 Encoding and Decoding Algorithm
- 5 Conclusion

Satellite Quantum Key Distribution (QKD)

Quantum Key Distribution.

- Cryptographic protocol involving components of quantum mechanics.
- Enables two parties to produce a shared random secret key known only to them.

Satellite Quantum Key Distribution (QKD)

Quantum Key Distribution.

Satellite QKD: share random key between satellite and ground station.

-
- [1] Peide Zhang et al. "Timing and synchronisation for high-loss free-space quantum communication with Hybrid de Bruijn Codes". In: *IET Quantum Communication* 2.3 (2021), pp. 80–89.
- [2] Isaac Khader et al. "Time synchronization over a free-space optical communication channel". In: *Optica* 5.12 (2018), pp. 1542–1548.

Satellite Quantum Key Distribution (QKD)

Quantum Key Distribution.

Satellite QKD: share random key between satellite and ground station.

A classical channel is used along to synchronise quantum channel^[1,2].

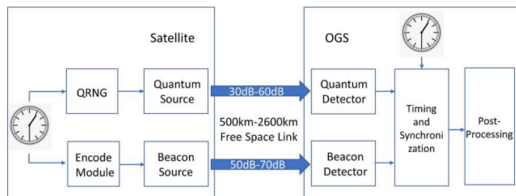


Figure 1: High-level satellite Quantum Key Distribution schematic^[1].

[1] Peide Zhang et al. "Timing and synchronisation for high-loss free-space quantum communication with Hybrid de Bruijn Codes". In: *IET Quantum Communication* 2.3 (2021), pp. 80–89.

[2] Isaac Khader et al. "Time synchronization over a free-space optical communication channel". In: *Optica* 5.12 (2018), pp. 1542–1548.

Zhang et al.^[1]: de Bruijn based timing-synchronization system (dBTS).

Zhang et al.^[1]: de Bruijn based timing-synchronization system (dBTS).

Encode



- Linear feedback shift register (LFSR):
generate an order k de Bruijn sequence.

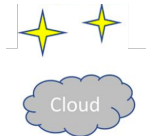
Zhang et al.^[1]: de Bruijn based timing-synchronization system (dBTS).

Encode



- Linear feedback shift register (LFSR): generate an order k de Bruijn sequence.

Noisy channel



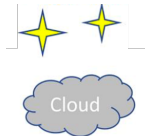
Zhang et al.^[1]: de Bruijn based timing-synchronization system (dBTS).

Encode



- Linear feedback shift register (LFSR): generate an order k de Bruijn sequence.

Noisy channel



Decode



- Look-up table: locate the position of a length k subsequence in the whole sequence.

Zhang et al.^[1]: de Bruijn based timing-synchronization system (dBTS).

Transmit a modulated sequence

- **Constraint:** avoid long period of no pulse .
- **Requirement:** positioning sequence.
- **Method:** de Bruijn sequence, pulse modulation:
 $1 \rightarrow \text{on} - \text{on}$, $0 \rightarrow \text{on} - \text{off}$, called Hybrid de Bruijn (HdB) sequence.

Zhang et al.^[1]: de Bruijn based timing-synchronization system (dBTS).

Transmit a modulated sequence

- **Constraint**: avoid long period of no pulse .
- **Requirement**: positioning sequence.
- **Method**: de Bruijn sequence, pulse modulation:
1 \rightarrow on – on, 0 \rightarrow on – off, called Hybrid de Bruijn (HdB) sequence.

Drawback

- Rate = 0.5.
- Encode: LFSR (prerequisite: suitable primitive polynomial).
- Decode: use look-up table (exponential complexity).

Propose a new combinatorial object RdB

- Can be encoded and decoded efficiently.
- Can replace HdB sequence: higher rate and maximal asymptotic rate, more general and adaptive.

Determine the maximal length of RdB

- Explicit formula.

Encoding and Decoding Algorithm

- Encoder: Constant amortized time per symbol.
- Decoder: Sub-linear time with respect to the length of the sequence.

Table of Contents

- 1 Introduction
- 2 Run length limited de Bruijn sequences
- 3 Rate and Maximal Asymptotic Rate
- 4 Encoding and Decoding Algorithm
- 5 Conclusion

De Bruijn Sequence (Positioning Sequence)

1 De Bruijn sequence:

- A cyclic binary de Bruijn of order k is a length 2^k sequence such that each length k string appears exactly once.
- Example: A de Bruijn sequence of order 4.
Cyclic : 0000100110101111.
Acyclic : 0000100110101111000.
- De Bruijn sequence \equiv longest simple path in de Bruijn graph (Eulerian cycle).

De Bruijn Sequence (Positioning Sequence)

1 De Bruijn sequence:

- A cyclic binary de Bruijn of order k is a length 2^k sequence such that each length k string appears exactly once.
- Example: A de Bruijn sequence of order 4.
Cyclic : 0000100110101111.
Acyclic : 0000100110101111000.
- De Bruijn sequence \equiv longest simple path in de Bruijn graph (Eulerian cycle).

2 De Bruijn graph of order k , G_k :

- Each vertex is labeled by a sequence of length $k - 1$.
- A directed edge from $\mathbf{x} = [x_0x_1 \dots x_{k-2}]$ to $\mathbf{y} = [y_0y_1 \dots y_{k-2}]$
 $\Leftrightarrow x_1x_2 \dots x_{k-2} = y_0y_1 \dots y_{k-3}$.



De Bruijn Sequence (Positioning Sequence)

1 De Bruijn sequence:

- A cyclic binary de Bruijn of order k is a length 2^k sequence such that each length k string appears exactly once.
- Example: A de Bruijn sequence of order 4.
Cyclic : 0000100110101111.
Acyclic : 0000100110101111000.
- De Bruijn sequence \equiv longest simple path in de Bruijn graph (Eulerian cycle).

2 De Bruijn graph of order k , G_k :

- Each vertex is labeled by a sequence of length $k - 1$.
- A directed edge from $\mathbf{x} = [x_0x_1 \dots x_{k-2}]$ to $\mathbf{y} = [y_0y_1 \dots y_{k-2}]$
 $\Leftrightarrow x_1x_2 \dots x_{k-2} = y_0y_1 \dots y_{k-3}$.

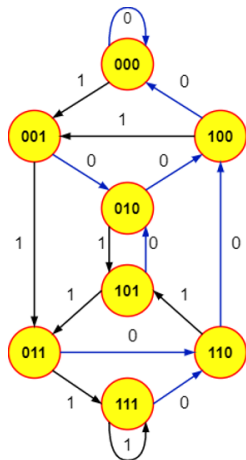


Figure 2: de Bruijn graph of order 4.



Run Length Limited de Bruijn (RdB) sequences

Definition 1

A (k, s) -RdB sequence: a de Bruijn sequence of order k containing at most s consecutive bit 0's.

Trivial cases:

- $s \geq k$: original de Bruijn sequence.
- $s = k - 1$: remove 0^{k-1} in the de Bruijn graph.

\Rightarrow Interested in:

- $s < k - 1$: eliminate vertices with more than s consecutive bit 0's.

Run Length Limited de Bruijn (RdB) sequences

Definition 1

A (k, s) -RdB sequence: a de Bruijn sequence of order k containing at most s consecutive bit 0's.

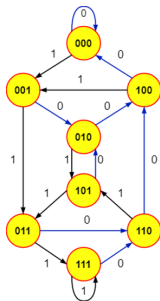


Figure 3: de Bruijn graph of order 4.

Run Length Limited de Bruijn (RdB) sequences

Definition 1

A (k, s) -RdB sequence: a de Bruijn sequence of order k containing at most s consecutive bit 0's.

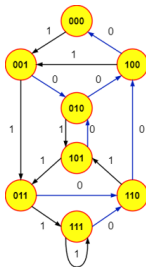
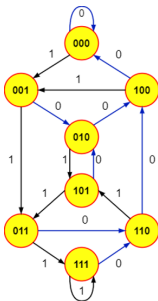


Figure 3: de Bruijn graph of order 4.

Figure 4: $(4, 3)$ -RdB graph.

Run Length Limited de Bruijn (RdB) sequences

Definition 1

A (k, s) -RdB sequence: a de Bruijn sequence of order k containing at most s consecutive bit 0's.

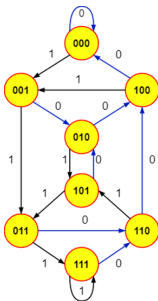


Figure 4: $(4, 3)$ -RdB graph.

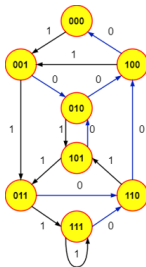


Figure 5: $(4, 1)$ -RdB graph.

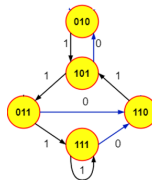


Figure 3: de Bruijn graph of order 4.

Maximal length of (k, s) -RdB sequence

Given k, s . Notations:

- $\ell(k, s)$: maximal length of simple path (k, s) -RdB graph.
- $N(k, s)$: maximal length of (k, s) -RdB sequences $(= \ell(k, s) + k - 1)$.

Theorem 2

Given k, s . Then:

$$\ell(k, s) = |W(k, s)| - \left(\sum_{i=0}^C (s - i) |W(k - s - i - 3, s)| - s \right) \quad (1)$$

where :

- $C = \min(s - 1, k - s - 2)$.
- $W(n, s)$: set of length n sequences containing at most s consecutive bit 0's.



Maximal length of (k, s) -RdB sequence

Lemma 3

$$\ell(k, s) \leq |W(k, s)| - \left(\sum_{i=0}^C (s-i) |W(k-s-i-3, s)| - s \right)$$

Observe:

- Vertices: balanced, right-unbalanced, left-unbalanced.
- Number of left(right)-unbalanced vertices of form $0^s 1 \dots 10^i$ is $|W(k-i-s-3, s)|$ with $i \leq C = \min(s-1, k-s-2)$.

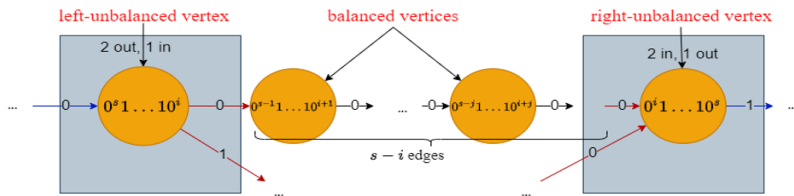


Figure 6: Vertices in RdB graphs.

Table of Contents

- 1 Introduction
- 2 Run length limited de Bruijn sequences
- 3 Rate and Maximal Asymptotic Rate**
- 4 Encoding and Decoding Algorithm
- 5 Conclusion

Definition of rate

Recall $N(k, s) = \ell(k, s) + k - 1$. Given k, s :

- Let $\mathbf{x}_{k,s}$: a (k, s) -RdB sequence. The (information) rate of $\mathbf{x}_{k,s}$:

$$R_{\mathbf{x}_{k,s}} = \frac{\log(|\mathbf{x}_{k,s}|)}{k} \quad (2)$$

- Maximal rate of (k, s) -RdB sequences:

$$R_{k,s} = \frac{\log(N(k, s))}{k} \quad (3)$$

- Maximal asymptotic rate of (k, s) -RdB sequences:

$$R_s = \lim_{k \rightarrow \infty} \frac{\log(N(k, s))}{k} \quad (4)$$

Definition of rate

Recall $N(k, s) = \ell(k, s) + k - 1$. Given k, s :

- Let $\mathbf{x}_{k,s}$: a (k, s) -RdB sequence. The (information) rate of $\mathbf{x}_{k,s}$:

$$R_{\mathbf{x}_{k,s}} = \frac{\log(|\mathbf{x}_{k,s}|)}{k} \quad (2)$$

- Maximal rate of (k, s) -RdB sequences:

$$R_{k,s} = \frac{\log(N(k, s))}{k} \quad (3)$$

- Maximal asymptotic rate of (k, s) -RdB sequences:

$$R_s = \lim_{k \rightarrow \infty} \frac{\log(N(k, s))}{k} \quad (4)$$

Definition of rate

Recall $N(k, s) = \ell(k, s) + k - 1$. Given k, s :

- Let $\mathbf{x}_{k,s}$: a (k, s) -RdB sequence. The (information) rate of $\mathbf{x}_{k,s}$:

$$R_{\mathbf{x}_{k,s}} = \frac{\log(|\mathbf{x}_{k,s}|)}{k} \quad (2)$$

- Maximal rate of (k, s) -RdB sequences:

$$R_{k,s} = \frac{\log(N(k, s))}{k} \quad (3)$$

- Maximal asymptotic rate of (k, s) -RdB sequences:

$$R_s = \lim_{k \rightarrow \infty} \frac{\log(N(k, s))}{k} \quad (4)$$

Maximal Asymptotic Rate

Theorem 4

$$R_1 = \log \left(\frac{1 + \sqrt{5}}{2} \right) = 0.6942$$

which is larger than 0.5, rate of HdB sequences.

Theorem 5

$$R_s = \log(|\omega|)$$

where ω is the root of equation: $x^{s+1} - x^s - \dots - x - 1 = 0$ such that $|\omega|$ is the largest.

Proof: Approximation: $|W(k, s)| \approx a |\omega|^n$. Hence:

$$N(k, s) \approx |\omega|^{k-s-2} \left(\sum_{i=0}^s a(i+1) |w|^{s-i} + \frac{s+k-1}{|w|^{k-s-2}} \right)$$



Maximal Asymptotic Rate

Theorem 4

$$R_1 = \log \left(\frac{1 + \sqrt{5}}{2} \right) = 0.6942$$

which is larger than 0.5, rate of HdB sequences.

Theorem 5

$$R_s = \log(|\omega|)$$

where ω is the root of equation: $x^{s+1} - x^s - \dots - x - 1 = 0$ such that $|\omega|$ is the largest.

Proof: Approximation: $|W(k, s)| \approx a |\omega|^n$. Hence:

$$\Rightarrow R_s = \lim_{n \rightarrow \infty} \frac{(k - s - 2) \log(|\omega|)}{k} = \log(|\omega|).$$

Table of Contents

- 1 Introduction
- 2 Run length limited de Bruijn sequences
- 3 Rate and Maximal Asymptotic Rate
- 4 Encoding and Decoding Algorithm**
- 5 Conclusion

- Based on lexicographic minimal de Bruijn sequence (granddaddy by Knuth):
Example with order 5: 0 00001 00011 00101 00111 01011 01111 1
→ Cut the sequence at the "right place".
- Complexity = Complexity to generate granddaddy sequence.
= Constant amortized time per symbol^[3].
- The correctness can be proved easily.

Cut at $u = 0^{s+1}1^{k-s-1}$.

Example $k = 5, s = 2$:

[3] Frank Ruskey, Carla Savage, and Terry Min Yih Wang. "Generating necklaces". In *Journal of Algorithms* 13.3 (1992), pp. 414–430.

- Based on lexicographic minimal de Bruijn sequence (granddaddy by Knuth):

Example with order 5: 0 00001 00011 00101 00111 01011 01111 1

→ Cut the sequence at the "right place".

- Complexity = Complexity to generate granddaddy sequence.
= Constant amortized time per symbol^[3].
- The correctness can be proved easily.

Cut at $u = 0^{s+1}1^{k-s-1}$.

Example $k = 5, s = 2$:

[3] Frank Ruskey, Carla Savage, and Terry Min Yih Wang. "Generating necklaces". In *Journal of Algorithms* 13.3 (1992), pp. 414–430.

- Based on lexicographic minimal de Bruijn sequence (granddaddy by Knuth):

Example with order 5: 0 00001 00011 00101 00111 01011 01111 1

→ Cut the sequence at the "right place".

- Complexity = Complexity to generate granddaddy sequence.
= Constant amortized time per symbol^[3].
- The correctness can be proved easily.

Cut at $u = 0^{s+1}1^{k-s-1}$.

Example $k = 5, s = 2$:

[3] Frank Ruskey, Carla Savage, and Terry Min Yih Wang. "Generating necklaces". *Journal of Algorithms* 13.3 (1992), pp. 414–430.

- Based on lexicographic minimal de Bruijn sequence (granddaddy by Knuth):
Example with order 5: 0 00001 00011 00101 00111 01011 01111 1
→ Cut the sequence at the "right place".
- Complexity = Complexity to generate granddaddy sequence.
= Constant amortized time per symbol^[3].
- The correctness can be proved easily.

Cut at $u = 0^{s+1}1^{k-s-1}$.

Example $k = 5, s = 2$:

0 00001 00011 00101 00111 01011 01111 1

[3] Frank Ruskey, Carla Savage, and Terry Min Yih Wang. "Generating necklaces". *Journal of Algorithms* 13.3 (1992), pp. 414–430.

- Based on lexicographic minimal de Bruijn sequence (granddaddy by Knuth):

Example with order 5: 0 00001 00011 00101 00111 01011 01111 1
→ Cut the sequence at the "right place".

- Complexity = Complexity to generate granddaddy sequence.
= Constant amortized time per symbol^[3].
- The correctness can be proved easily.

Cut at $u = 0^{s+1}1^{k-s-1}$.

Example $k = 5, s = 2$:

0 00001 0|0011 00101 00111 01011 01111 1

[3] Frank Ruskey, Carla Savage, and Terry Min Yih Wang. "Generating necklaces". *Journal of Algorithms* 13.3 (1992), pp. 414–430.

- Based on lexicographic minimal de Bruijn sequence (granddaddy by Knuth):

Example with order 5: 0 00001 00011 00101 00111 01011 01111 1
→ Cut the sequence at the "right place".

- Complexity = Complexity to generate granddaddy sequence.
= Constant amortized time per symbol^[3].
- The correctness can be proved easily.

Cut at $u = 0^{s+1}1^{k-s-1}$.

Example $k = 5, s = 2$:

0 00001 0|0011 00101 00111 01011 01111 1 00

[3] Frank Ruskey, Carla Savage, and Terry Min Yih Wang. "Generating necklaces". In *Journal of Algorithms* 13.3 (1992), pp. 414–430.

Algorithm 1: Encode (k,s) -RdB

Input : k , and descending ordered set $\mathcal{L}^{(n)}$.

Output: (k,s) -RLL dBs

$\mathbf{w} \leftarrow \text{emptystring}$

for $\lambda \in \mathcal{L}^{(n)}$ **do**

$\mathbf{w}.\text{prepend}(\lambda)$

if $\lambda == 0^{s+1}1^{k-s-1}$ **then**

$\mathbf{w} = \mathbf{w}[2, \ell]0^s$

break

return \mathbf{w}

$\mathcal{L}^{(n)}$: generated by FKM algorithm^[4,5].

[4] [Harold Fredricksen and James Maiorana](#). "Necklaces of beads in k colors and k -ary de Bruijn sequences". In: *Discrete Mathematics* 23.3 (1978), pp. 207–210.

[5] [Harold Fredricksen and Irving J Kessler](#). "An algorithm for generating necklaces of beads in two colors". In: *Discrete mathematics* 61.2-3 (1986), pp. 181–188.



The Optimality of The Encoder

- We're proving that our encoder generate the longest (k, s) -RdB sequences.

Example 6 ($k = 5, s = 2$)

0 00001 0 | 0011 00101 00111 01011 01111 100
 this length= $N(k,s)$?

The Optimality of The Encoder

- We're proving that our encoder generate the longest (k, s) -RdB sequences.

Example 6 ($k = 5, s = 2$)

0 00001 0 | 0011 00101 00111 01011 01111 100
this length = $N(k, s)$?

Equivalent to:

0 00001 0 | 0011 00101 00111 01011 01111 100
How long is this sequence? ($=X$)

We'll prove that $X = N(k, s) - s$



The Optimality of The Encoder

$\langle \mathbf{v} \rangle$ minimal rotation of \mathbf{v} (exp: $\langle 110 \rangle = 011$, $\langle 1001 \rangle = 0011$).

$S(\mathbf{v}) = \{ \mathbf{x} \in \Sigma^{|\mathbf{v}|} : \langle \mathbf{x} \rangle < \mathbf{v} \}$.

$\mathcal{L}^{(n)}$: set of Lyndon words whose length is a divisor of n .

Lemma 7 (Lemma 29^[6])

Let \mathbf{v} be a Lyndon word. Define $\mathcal{L}(\mathbf{v}) = \{ \lambda \in \mathcal{L}^k : \lambda^{\frac{k}{|\lambda|}} \leq \mathbf{v} \}$ to be the set of all Lyndon words smaller than \mathbf{v} whose length is the divisor of k . Then: $\sum_{\lambda \in \mathcal{L}(\mathbf{v})} |\lambda| = |S(\mathbf{v})|$.

length= $|S(\mathbf{u})|$
0 00001 0|0011 00101 00111 01011 01111 1
length needs calculating

[6] Tomasz Kociumaka, Jakub Radoszewski, and Wojciech Rytter. "Efficient ranking of Lyndon words and decoding lexicographically minimal de Bruijn sequence". In: *SIAM Journal on Discrete Mathematics* 30.4 (2016), pp. 2027–2046.

The Optimality of The Encoder

$$|S(\mathbf{u})| = 1 + \sum_{t=M}^k (k-t+1) |C(t-2, s)| + \sum_{t=1}^{k-s} (k-t+1) A_t \quad (5)$$

where $|C(k, s)| = 2^k - |W(k, s)|$, $A_t = 2^{t-2}$, $M = \max(s+2, k-s+1)$

The Optimality of The Encoder

For $s = 1$, $|S(\mathbf{u})| = 2^k - (|W(k, 1)| - |W(k - 4, 1)|)$

The Optimality of The Encoder

For $s = 1$, $|S(\mathbf{u})| = 2^k - (|W(k, 1)| - |W(k - 4, 1)|)$

$k = 5, s = 1$

$$\overbrace{0 \ 00001 \ 00011 \ 00101 \ 0 \ 0111}^{2^k - (|W(k, 1)| - |W(k - 4, 1)|)} \ 01011 \ 01111 \ 1$$

The Optimality of The Encoder

For $s = 1$, $|S(\mathbf{u})| = 2^k - (|W(k, 1)| - |W(k - 4, 1)|)$

$k = 5, s = 1$

$$\begin{array}{c} \overbrace{0 \ 00001 \ 00011 \ 00101}^{2^k - (|W(k, 1)| - |W(k - 4, 1)|)} \ 0 \underbrace{|0111 \ 01011 \ 01111 \ 1}_{|W(k, 1)| - |W(k - 4, 1)| + k} \end{array}$$

The Optimality of The Encoder

For $s = 1$, $|S(\mathbf{u})| = 2^k - (|W(k, 1)| - |W(k - 4, 1)|)$

$k = 5, s = 1$

$$\begin{array}{c}
 \overbrace{0 \ 00001 \ 00011 \ 00101 \ 0|0111 \ 01011 \ 01111 \ 1}^{2^k - (|W(k, 1)| - |W(k - 4, 1)|)} \\
 \underbrace{\hspace{10em}}_{|W(k, 1)| - |W(k - 4, 1)| + k} \\
 \hspace{10em} = N(k, 1)
 \end{array}$$

- ① Based on the decoder \mathcal{D}_{KRR} proposed by Kociumaka, Radoszewski, and Rytter^[6].
- ② Based on the observation:
 - $i = \mathcal{D}_{KRR}(\mathbf{u} = 0^{s+1}1^{k-s-1})$: position of \mathbf{u} in granddaddy sequence \mathbf{x} of order k .
 - Location of \mathbf{v} ($|\mathbf{v}| = k$) in encoded sequence = Location of \mathbf{v} in $\mathbf{x} - i$.
 $= \mathcal{D}_{KRR}(\mathbf{v}) - i$.
- ③ Complexity = Complexity of \mathcal{D}_{KRR} .

- ① Based on the decoder \mathcal{D}_{KRR} proposed by Kociumaka, Radoszewski, and Rytter^[6].
- ② Based on the observation:
 - $i = \mathcal{D}_{KRR}(\mathbf{u} = 0^{s+1}1^{k-s-1})$: position of \mathbf{u} in granddaddy sequence \mathbf{x} of order k .
 - Location of \mathbf{v} ($|\mathbf{v}| = k$) in encoded sequence = Location of \mathbf{v} in $\mathbf{x} - i$.
 $= \mathcal{D}_{KRR}(\mathbf{v}) - i$.
- ③ Complexity = Complexity of \mathcal{D}_{KRR} .

Exp: $k = 5, s = 1$

then $\mathbf{u} = 00111$

say: $\mathbf{v} = 10111$

- ① Based on the decoder \mathcal{D}_{KRR} proposed by Kociumaka, Radoszewski, and Rytter^[6].
- ② Based on the observation:
 - $i = \mathcal{D}_{KRR}(\mathbf{u} = 0^{s+1}1^{k-s-1})$: position of \mathbf{u} in granddaddy sequence \mathbf{x} of order k .
 - Location of \mathbf{v} ($|\mathbf{v}| = k$) in encoded sequence = Location of \mathbf{v} in $\mathbf{x} - i$.
 $= \mathcal{D}_{KRR}(\mathbf{v}) - i$.
- ③ Complexity = Complexity of \mathcal{D}_{KRR} .

Exp: $k = 5, s = 1$

then $\mathbf{u} = 00111$

0 00001 00011 00101 00111 01011 01111 1 0

say: $\mathbf{v} = 10111$

- ① Based on the decoder \mathcal{D}_{KRR} proposed by Kociumaka, Radoszewski, and Rytter^[6].
- ② Based on the observation:
 - $i = \mathcal{D}_{KRR}(\mathbf{u} = 0^{s+1}1^{k-s-1})$: position of \mathbf{u} in granddaddy sequence \mathbf{x} of order k .
 - Location of \mathbf{v} ($|\mathbf{v}| = k$) in encoded sequence = Location of \mathbf{v} in $\mathbf{x} - i$.
 $= \mathcal{D}_{KRR}(\mathbf{v}) - i$.
- ③ Complexity = Complexity of \mathcal{D}_{KRR} .

Exp: $k = 5, s = 1$

then $\mathbf{u} = 00111$

say: $\mathbf{v} = 10111$

0 00001 00011 00101 00111 01011 01111 1 0
 i

- ① Based on the decoder \mathcal{D}_{KRR} proposed by Kociumaka, Radoszewski, and Rytter^[6].
- ② Based on the observation:
 - $i = \mathcal{D}_{KRR}(\mathbf{u} = 0^{s+1}1^{k-s-1})$: position of \mathbf{u} in granddaddy sequence \mathbf{x} of order k .
 - Location of \mathbf{v} ($|\mathbf{v}| = k$) in encoded sequence = Location of \mathbf{v} in $\mathbf{x} - i$.
 $= \mathcal{D}_{KRR}(\mathbf{v}) - i$.
- ③ Complexity = Complexity of \mathcal{D}_{KRR} .

Exp: $k = 5, s = 1$

then $\mathbf{u} = 00111$

say: $\mathbf{v} = 10111$

0 00001 00011 00101 00111 0101 1 0111 1 1 0
 i j

- ① Based on the decoder \mathcal{D}_{KRR} proposed by Kociumaka, Radoszewski, and Rytter^[6].
- ② Based on the observation:
 - $i = \mathcal{D}_{KRR}(\mathbf{u} = 0^{s+1}1^{k-s-1})$: position of \mathbf{u} in granddaddy sequence \mathbf{x} of order k .
 - Location of \mathbf{v} ($|\mathbf{v}| = k$) in encoded sequence = Location of \mathbf{v} in $\mathbf{x} - i$.
 $= \mathcal{D}_{KRR}(\mathbf{v}) - i$.
- ③ Complexity = Complexity of \mathcal{D}_{KRR} .

Exp: $k = 5, s = 1$

then $\mathbf{u} = 00111$

say: $\mathbf{v} = 10111$

0	00001	00011	00101	00111	0101	1 0111	1	1	0	
				i		j				
				0111	0101	1	0111	1	1	0

- ① Based on the decoder \mathcal{D}_{KRR} proposed by Kociumaka, Radoszewski, and Rytter^[6].
- ② Based on the observation:
 - $i = \mathcal{D}_{KRR}(\mathbf{u} = 0^{s+1}1^{k-s-1})$: position of \mathbf{u} in granddaddy sequence \mathbf{x} of order k .
 - Location of \mathbf{v} ($|\mathbf{v}| = k$) in encoded sequence = Location of \mathbf{v} in $\mathbf{x} - i$.
 $= \mathcal{D}_{KRR}(\mathbf{v}) - i$.
- ③ Complexity = Complexity of \mathcal{D}_{KRR} .

Exp: $k = 5, s = 1$

then $\mathbf{u} = 00111$

say: $\mathbf{v} = 10111$

0	00001	00011	00101	00111	0101	1 0111	1	1	0
				i		j			
				0111	0101	1 0111	1	1	0
						$j-i$			

Denote $c_{k,s}$ to be the encoded sequence.

Algorithm 2: Decode (k,s)-RdB $c_{k,s}$

Input : A word $\mathbf{v} = (v_1, \dots, v_k)$ of length k

Output: a is the location of \mathbf{v} in $c_{k,s}$

```
 $i \leftarrow \mathcal{D}_{KRR}(\mathbf{u}_{k,s});$   
if  $\mathbf{v} = 1^j 0^{k-j}$ , then  
   $\text{return } n - i + 1 - (k - j);$   
else  
   $\text{return } \mathcal{D}_{KRR}(\mathbf{v}) - i;$ 
```

Table of Contents

- 1 Introduction
- 2 Run length limited de Bruijn sequences
- 3 Rate and Maximal Asymptotic Rate
- 4 Encoding and Decoding Algorithm
- 5 Conclusion**

Summary

A new combinatorial structure

- Run length limited de Bruijn sequences (RdB).
- Encode and decode efficiently.

Length and rate

- Explicit formula for maximal length and maximal asymptotic rate of RdB sequences.

Encoding and decoding algorithm

- Encode in constant amortized time per symbol.
- Decode in polynomial time (sub-linear with respect to the length of the whole sequence).

More Constraints

- Weight constraint.
- Locally constraint.
- Extend the size of alphabet.

Properties

- Lexicographic minimal RdB sequences ?
- How many sequences of the same size ?

My publications during the time doing this thesis:

- Yeow Meng Chee, Duc Tu Dao, **Tien Long Nguyen**, Duy Hoang Ta, Van Khu Vu. "Run Length Limited de Bruijn Sequences for Quantum Communications". In proceeding of IEEE International Symposium on Information Theory, 2022.
- Tran Ba Trung, Lijun Chang, **Nguyen Tien Long**, Kai Yao, Huynh Thi Thanh Binh. "Verification-Free Approaches to Efficient Locally Densest Subgraph Discovery". Accepted as paper at The 39th IEEE International Conference on Data Engineering, 2023.

- [1] Peide Zhang et al. “Timing and synchronisation for high-loss free-space quantum communication with Hybrid de Bruijn Codes”. In: *IET Quantum Communication* 2.3 (2021), pp. 80–89.
- [2] Isaac Khader et al. “Time synchronization over a free-space optical communication channel”. In: *Optica* 5.12 (2018), pp. 1542–1548.
- [3] Frank Ruskey, Carla Savage, and Terry Min Yih Wang. “Generating necklaces”. In: *Journal of Algorithms* 13.3 (1992), pp. 414–430.
- [4] Harold Fredricksen and James Maiorana. “Necklaces of beads in k colors and k -ary de Bruijn sequences”. In: *Discrete Mathematics* 23.3 (1978), pp. 207–210.
- [5] Harold Fredricksen and Irving J Kessler. “An algorithm for generating necklaces of beads in two colors”. In: *Discrete mathematics* 61.2-3 (1986), pp. 181–188.

- [6] Tomasz Kociumaka, Jakub Radoszewski, and Wojciech Rytter.
“Efficient ranking of Lyndon words and decoding lexicographically
minimal de Bruijn sequence”. In: *SIAM Journal on Discrete
Mathematics* 30.4 (2016), pp. 2027–2046.