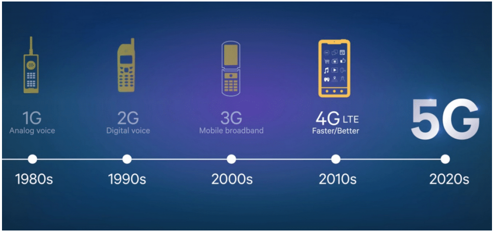
Neural Dynamic Successive-Cancellation Flip Decoding of Polar Codes

Nghia Doan[1], Seyyed Ali Hashemi[2], Furkan Ercan[1], Thibaud Tonnellier[1], and Warren Gross[1]

[1]McGill University, Québec, Canada
[2]Stanford University, California, USA

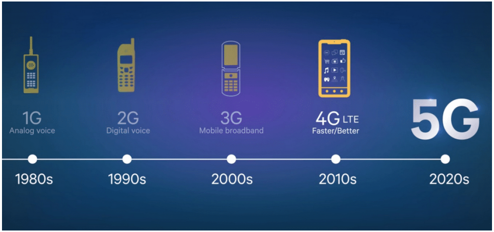
IEEE SiPS
Nanjing, China
Oct 21, 2019

Background



- Polar codes: selected for the eMBB control channel in 5G
- Cyclic redundancy check (CRC) is concatenated with polar codes in 5G for error detection
- Successive Cancellation List (SCL) for CRC-Polar concatenated codes
  - Good error correction performance
  - Complexity increases when list size increases

Background



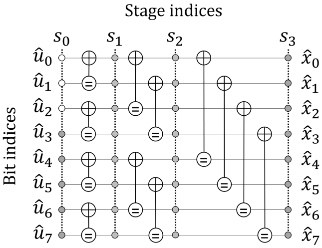
- Dynamic Successive Cancellation Flip (DSCF):
  - Comparable error-correction performance with SCL
  - Average decoding latency approaches that of SC at practical signal-to-noise ratio (SNR) regimes
  - **Costly transcendental computations**

This talk

- Propose a bit-flipping metric that only requires additions
- Propose a novel training framework for the decoder's parameter

Polar codes

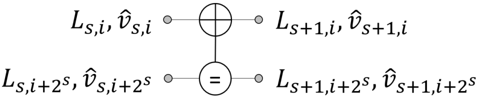
- Introduced by Arıkan [1] in 2009
- $\mathcal{P}(N, K)$, N: code length, K: message length
- Code construction: based on polarization phenomenon
  - K most reliable channels: information bits
  - $(N-K)$ least reliable channels: frozen bits



$\mathcal{P}(8, 5)$ with u_0, u_1, and u_2 are frozen bits

[1] E. Arıkan, "Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels", IEEE Trans. on Info. Theory, vol. 55, no. 7, pp. 3051–3073, July 2009.

Successive Cancellation (SC) Decoding



- Soft-value update rules:

$$\begin{cases} L_{s,i} = f(L_{s+1,i}, L_{s+1,i+2^s}) \\ L_{s,i+2^s} = g(L_{s+1,i}, L_{s+1,i+2^s}, \hat{\nu}_{s,i}), \end{cases}$$

where

$$\begin{cases} f(a,b) = \min(|a|,|b|)\,\mathrm{sgn}(a)\,\mathrm{sgn}(b) \\ g(a,b,c) = b + (1-2c)a, \end{cases}$$

- Hard-value update rules:

$$\begin{cases} \hat{\nu}_{s+1,i} = \hat{\nu}_{s,i} \oplus \hat{\nu}_{s,i+2^s} \\ \hat{\nu}_{s+1,i+2^s} = \hat{\nu}_{s,i+2^s}, \end{cases}$$

where

$$\hat{u}_i = \hat{\nu}_{0,i} = \begin{cases} 0 & \text{if } u_i \text{ is frozen,} \\ \frac{1-\mathrm{sgn}(L_{0,i})}{2} & \text{otherwise.} \end{cases}$$

[1] E. Arıkan, ”Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels”, IEEE Trans. on Info. Theory, vol. 55, no. 7, pp. 3051–3073, July 2009.

Successive Cancellation Flip (SCF) Decoding

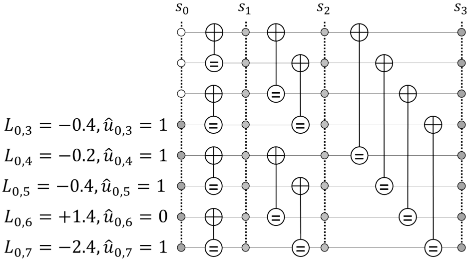
- Given that SC decoding fails the CRC verification
- SCF [2] flips the hard-decision of an information bit, then performs SC decoding again
- Flipping index $i = \arg\min_{\forall i \in \mathcal{A}} |L_{0,i}|$, where $\mathcal{A}$ is the information set [2]

[2] Orion, et al., "A low-complexity improved successive cancellation decoder for polar codes." IEEE Asilomar Conf. on Sign., Sys. and Comp, 2014

Successive Cancellation Flip (SCF) Decoding

Example:

- An all-zero codeword is sent
- Flipping position $i = \arg\min_{\forall i \in \mathcal{A}} |L_{0,i}| = 4$

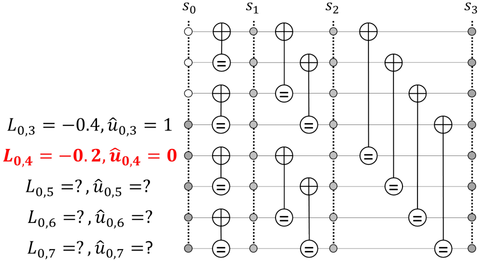


[2] Orion, et al., "A low-complexity improved successive cancellation decoder for polar codes." IEEE Asilomar Conf. on Sign., Sys. and Comp, 2014

Successive Cancellation Flip (SCF) Decoding

Example:

- An all-zero codeword is sent
- Flipping position $i = \arg\min_{\forall i \in \mathcal{A}} |L_{0,i}| = 4$



---

[2] Orion, et al., "A low-complexity improved successive cancellation decoder for polar codes." IEEE Asilomar Conf. on Sign., Sys. and Comp, 2014

Dynamic SC-Flip (DSCF) Decoding

- Propose a bit-flipping metric for high-order error bits

[3] Chandesris et al., "Dynamic-SCFlip decoding of polar codes." IEEE Transactions on Communications 66.6 (2018): 2333-2345.

Dynamic SC-Flip (DSCF) Decoding

- Propose a bit-flipping metric for high-order error bits
- Bit-flipping probability at the i-th information bit [3]:

$$Pr_i = \prod_{\forall j \in \mathcal{E}_\omega,\, j \leq i} \left(1 - p_j^*\right) \prod_{\forall j \in \mathcal{A} \setminus \mathcal{E}_\omega,\, j < i} p_j^* \quad (1)$$

where

- $\mathcal{E}_\omega$: the set of bit-flipping position at error order ω-th
- $p_j^* = Pr(\hat{u}_i = u_i | \hat{\boldsymbol{u}}_0^{i-1} = \boldsymbol{u}_0^{i-1}) \approx \frac{1}{1+\exp(-\alpha|L_{0,j}|)}$
- $\alpha > 0$ is a perturbation parameter

[3] Chandesris et al., "Dynamic-SCFlip decoding of polar codes." IEEE Transactions on Communications 66.6 (2018): 2333-2345.

Dynamic SC-Flip (DSCF) Decoding

- Propose a bit-flipping metric for high-order error bits
- Bit-flipping probability at the i-th information bit [3]:

$$Pr_i = \prod_{\forall j \in \mathcal{E}_\omega,\, j \leq i} \left(1 - p_j^*\right) \prod_{\forall j \in \mathcal{A} \setminus \mathcal{E}_\omega,\, j < i} p_j^* \tag{2}$$

  where

  - $\mathcal{E}_\omega$: the set of bit-flipping position at error order ω-th
  - $p_j^* = Pr(\hat{u}_i = u_i | \hat{\boldsymbol{u}}_0^{i-1} = \boldsymbol{u}_0^{i-1}) \approx \frac{1}{1+\exp(-\alpha|L_{0,j}|)}$
  - $\alpha > 0$ is a perturbation parameter

- Flipping position $i = \arg\max_{\forall i \in \mathcal{A}} Pr_i$

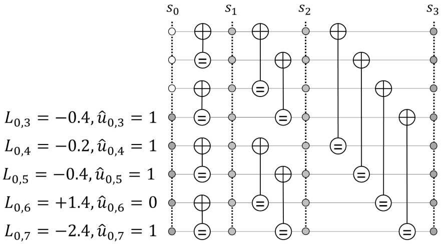
[3] Chandesris et al., "Dynamic-SCFlip decoding of polar codes." IEEE Transactions on Communications 66.6 (2018): 2333-2345.

Dynamic SC-Flip (DSCF) Decoding

Example:

- $p_i^* = \frac{1}{1+\exp(-\alpha|L_{0,i}|)}$
- $\alpha = 0.3$
- At $\omega = 1$:

i	p_i^*	$1 - p_i^*$	Pr_i
3	0.53	0.47	-
4	0.51	0.49	-
5	0.53	0.47	-
6	0.60	0.40	-
7	0.67	0.33	-

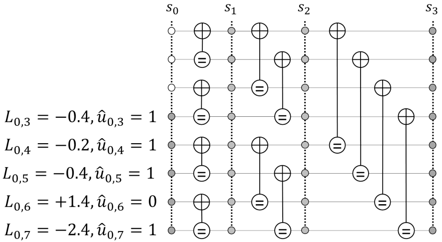


Dynamic SC-Flip (DSCF) Decoding

- At $\omega = 1$:

i	p_i^*	$1 - p_i^*$	Pr_i
3	0.53	0.47	0.47
4	0.51	0.49	0.26
5	0.53	0.47	0.13
6	0.60	0.40	0.06
7	0.67	0.33	0.03

- $Pr_3 = 1 - p_3^*$
- ...
- $Pr_7 =$ $p_3^* \times p_4^* \times p_5^* \times p_6^* \times (1 - p_7^*)$
- $i = \arg\max_{\forall i \in \mathcal{A}} Pr_i = 0$

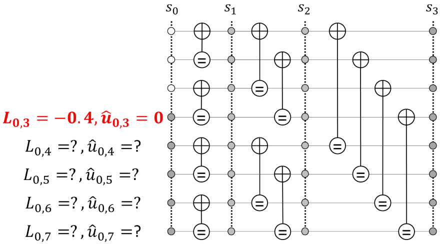


Dynamic SC-Flip (DSCF) Decoding

- At $\omega = 1$:

i	p_i^*	$1 - p_i^*$	Pr_i
3	**0.53**	**0.47**	**0.47**
4	0.51	0.49	0.26
5	0.53	0.47	0.13
6	0.60	0.40	0.06
7	0.67	0.33	0.03

- $Pr_3 = 1 - p_3^*$
- $\dots$
- $Pr_7 =$
  $p_3^* \times p_4^* \times p_5^* \times p_6^* \times (1 - p_7^*)$
- $i = \arg\max_{\forall i \in \mathcal{A}} Pr_i = 0$

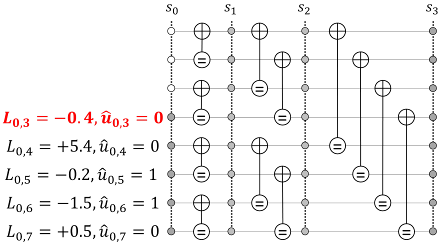


Dynamic SC-Flip (DSCF) Decoding

- At $\omega = 2$:

i	p_i^*	$1 - p_i^*$	Pr_i
3	**0.53**	**0.47**	**0.47**
4	0.83	0.17	0.08
5	0.51	0.49	0.19
6	0.61	0.39	0.07
7	0.46	0.54	0.06

- $Pr_4 = (1 - p_3^*) \times (1 - p_4^*)$
- ...
- $Pr_7 = (1 - p_3^*) \times p_4^* \times p_5^* \times p_6^* \times (1 - p_7^*)$
- $i = \arg\max_{\forall i \in \mathcal{A}} Pr_i = 5$

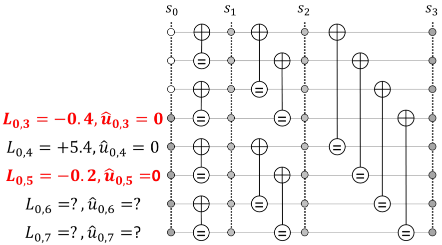


Dynamic SC-Flip (DSCF) Decoding

- At $\omega = 2$:

i	p_i^*	$1 - p_i^*$	Pr_i
3	**0.53**	**0.47**	**0.47**
4	0.83	0.17	0.08
5	**0.51**	**0.49**	**0.19**
6	0.61	0.39	0.07
7	0.46	0.54	0.06

- $Pr_4 = (1 - p_3^*) \times (1 - p_4^*)$
- ...
- $Pr_7 = (1 - p_3^*) \times p_4^* \times p_5^* \times p_6^* \times (1 - p_7^*)$
- $i = \arg\max_{\forall i \in \mathcal{A}} Pr_i = 5$



Dynamic SC-Flip (DSCF) Decoding-Problems

- To enable numerical stability

$$
\begin{aligned}
Q_i = -\frac{1}{\alpha}\ln(Pr_i) = \sum_{\substack{\forall j\in\mathcal{A}\\ j\leq i}} \frac{1}{\alpha}\ln\left(1+\exp\left(-\alpha|L_{0,j}|\right)\right) \\
+ \sum_{\substack{\forall j\in\mathcal{E}_\omega\\ j\leq i}} |L_{0,j}|.
\end{aligned}
\tag{3}
$$

- **Computing Q_i requires costly $\ln$ and $\exp$ functions!**

Dynamic SC-Flip (DSCF) Decoding-Problems

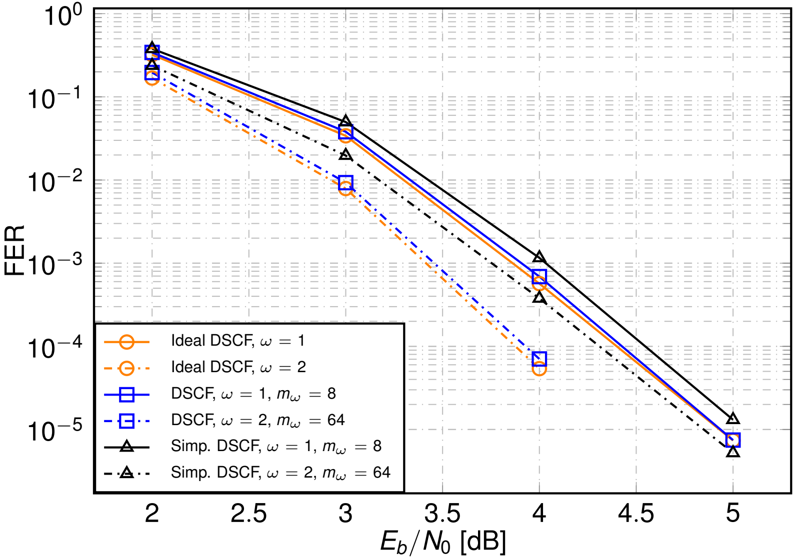
- Simplify Q_i using a conventional technique [4]:

$$\ln(1+\exp(x)) \approx \mathrm{ReLU}(x) \tag{4}$$

$$\tilde{Q}_i = \sum_{\substack{\forall j \in \mathcal{A} \\ j \leq i}} \frac{1}{\alpha} \mathrm{ReLU}\left(-\alpha |L_{0,j}|\right) + \sum_{\substack{\forall j \in \mathcal{E}_\omega \\ j \leq i}} |L_{0,j}| = \sum_{\substack{\forall j \in \mathcal{E}_\omega \\ j \leq i}} |L_{0,j}| \tag{5}$$

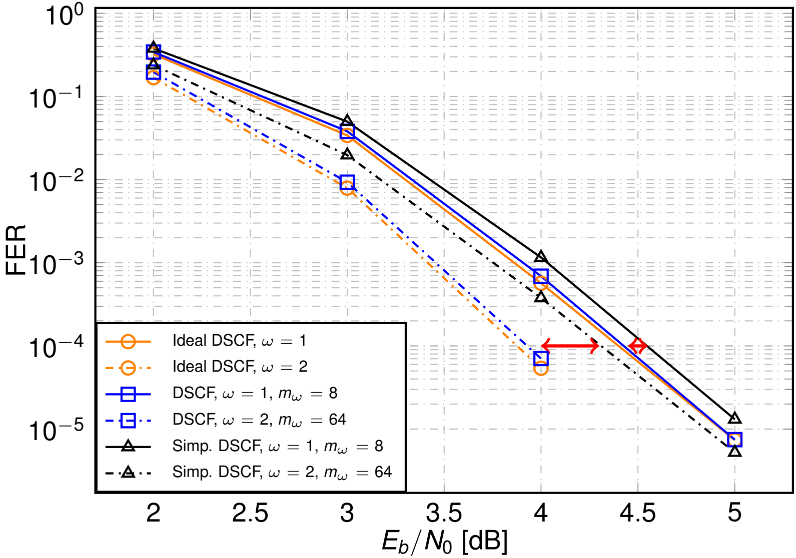
[4] Balatsoukas-Stimming et al., ”LLR-based successive cancellation list decoding of polar codes.” IEEE Trans. on Sig. Process. 63.19 (2015): 5165-5179.

Dynamic SC-Flip (DSCF) Decoding-Problems



$\mathcal{P}(256, 128) - C24$ used in 5G.

Dynamic SC-Flip (DSCF) Decoding-Problems



Simplifying DSCF hurts the FER performance!

Neural Dynamic SC-Flip (NDSCF) Decoding

- Introduce an additive perturbation parameter to estimate p_j^*

$$p_j^* \approx \frac{1}{1+\exp(\beta - |L_{0,j}|)} \tag{6}$$

instead of [3]

$$p_j^* \approx \frac{1}{1+\exp(-\alpha|L_{0,j}|)}$$

[3] Chandesris et al., "Dynamic-SCFlip decoding of polar codes." IEEE Transactions on Communications 66.6 (2018): 2333-2345.

Neural Dynamic SC-Flip (NDSCF) Decoding

- Introduce an additive perturbation parameter to estimate p_j^*

$$p_j^* \approx \frac{1}{1 + \exp(\beta - |L_{0,j}|)} \tag{7}$$

- The tailored bit-flipping metric under NDSCF decoding

$$Q_i^* = \sum_{\substack{\forall j \in \mathcal{A} \\ j \leq i}} \mathrm{ReLU}\left(\beta - |L_{0,j}|\right) + \sum_{\substack{\forall j \in \mathcal{E}_\omega \\ j \leq i}} |L_{0,j}| \tag{8}$$

Neural Dynamic SC-Flip (NDSCF) Decoding

- DSCF

$$Q_i = \sum_{\substack{\forall j \in \mathcal{A} \\ j \leq i}} \frac{1}{\alpha} \ln\left(1 + \exp\left(-\alpha |L_{0,j}|\right)\right) + \sum_{\substack{\forall j \in \mathcal{E}_\omega \\ j \leq i}} |L_{0,j}|.$$

- Simplified DSCF

$$\tilde{Q}_i = \sum_{\substack{\forall j \in \mathcal{E}_\omega \\ j \leq i}} |L_{0,j}|.$$

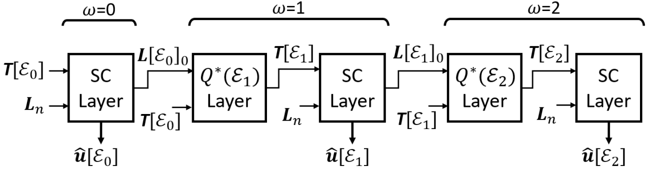
- NDSCF (proposed)

$$Q_i^* = \sum_{\substack{\forall j \in \mathcal{A} \\ j \leq i}} \mathrm{ReLU}\left(\beta - |L_{0,j}|\right) + \sum_{\substack{\forall j \in \mathcal{E}_\omega \\ j \leq i}} |L_{0,j}|$$

Decoder	$+$	$\times$	ln, exp
DSCF	✓	✓	✓
Simplified DSCF	✓		
NDSCF (proposed)	✓		

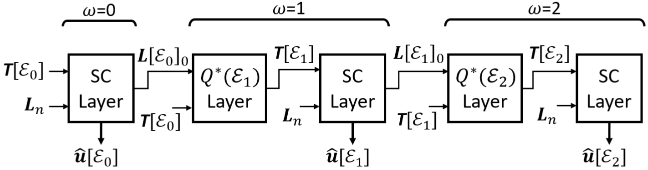
Neural Dynamic SC-Flip (NDSCF) Decoding

The optimization of β can be done using deep-learning techniques



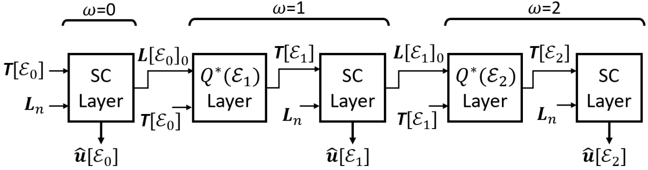
- SC layer: SC decoding
- $Q^*(\mathcal{E}_\omega)$: bit-flipping metric layer at error order ω-th
- $T[\mathcal{E}_\omega]$: bit-flipping vector at error order ω-th

Neural Dynamic SC-Flip (NDSCF) Decoding



- The proposed metric Q_i^* is independent from $\boldsymbol{u}$, hence all-zero codewords are used for training
- SC decoding can be implemented as a network layer to avoid the labeling task for the correct flipping position

Neural Dynamic SC-Flip (NDSCF) Decoding



Mapping from $\boldsymbol{Q}^*$ to $\boldsymbol{T}^*$:

- Inference:

  $\boldsymbol{Q}^* = \{Q_0^*, \dots, Q_{i_{min}}^*, \dots, Q_n^*\}$ where $Q_{i_{min}}^* = \min\{\boldsymbol{Q}^*\}$,
  thus $\boldsymbol{T}^* = \{T_0^* = 1, \dots, T_{i_{min}}^* = -1, \dots, T_n^* = 1\}$.

- Training:

  Let $Q_{i_{2ndmin}}^* = \min\{\boldsymbol{Q}^* \setminus Q_{i_{min}}^*\}$ and $Q_\tau = \frac{Q_{i_{2ndmin}}^* + Q_{i_{min}}^*}{2}$,
  $\boldsymbol{T}^* = \{T_0^* = \tanh(Q_0^* - Q_\tau), \dots, T_n^* = \tanh(Q_n^* - Q_\tau)\}$.

Neural Dynamic SC-Flip (NDSCF) Decoding

- Training loss:

$$\lambda = \sum_{\omega=1}^{2} l_\omega, \tag{9}$$

where

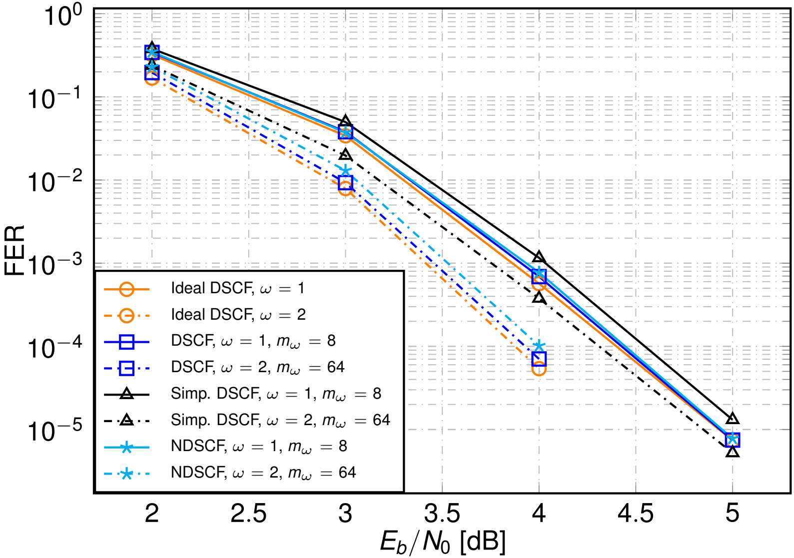
$$\begin{aligned} l_\omega &= \begin{cases} \left(\hat{u}[\mathcal{E}_\omega]_{t_\omega} - u_{t_\omega}\right)^2 & \text{if } \mathrm{CRC}(\hat{\boldsymbol{u}}[\mathcal{E}_\omega]) \neq 0,\ i^*_\omega \geq t_\omega, \\ 0 & \text{otherwise}, \end{cases} \\ &\approx \begin{cases} \frac{1}{\left(1+\exp\left(L[\mathcal{E}_\omega]_{0,t_\omega}\right)\right)^2} & \text{if } \mathrm{CRC}(\hat{\boldsymbol{u}}[\mathcal{E}_\omega]) \neq 0,\ i^*_\omega \geq t_\omega, \\ 0 & \text{otherwise}, \end{cases} \end{aligned} \tag{10}$$

t_ω: the first erroneous bit position of $\hat{\boldsymbol{u}}[\mathcal{E}_\omega]$
$u_{t_\omega} = 0$

- Optimized parameters:

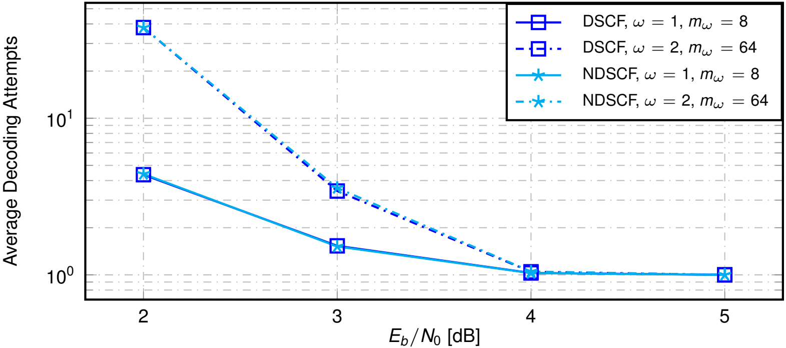
ω	1	2
β	2.801	2.196

Neural Dynamic SC-Flip (NDSCF) Decoding



$\mathcal{P}(256, 128) - C24$ used in 5G.

Neural Dynamic SC-Flip (NDSCF) Decoding



Average decoding attempts for $\mathcal{P}(256, 128) - C24$ used in 5G.

Conclusion

- Propose a simplified DSCF decoding algorithm using neural-network techniques
- Only additions are used to calculate the bit-flipping metric
- Incur negligible error-correction performance loss compared to DSCF decoding
- Preserve the same average decoding latency compared to SC decoding

Thank You!