Bit Error Robustness for Energy-Efficient DNN Accelerators

David Stutz

Nandhini Chandramoorthy

Matthias Hein

Bernt Schiele

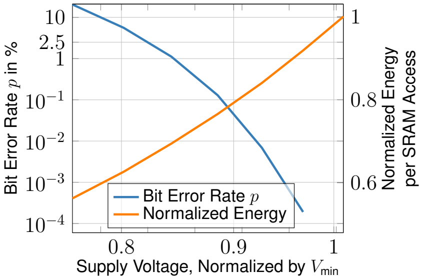
max planck institut informatik

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

IBM

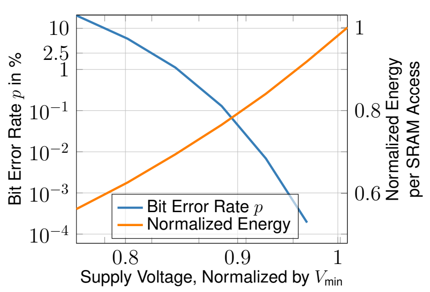
1-Minute Overview: Bit Error Robustness

Random **bit errors**:

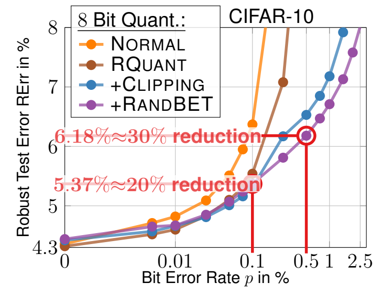


1-Minute Overview: Bit Error Robustness

Random **bit errors**:

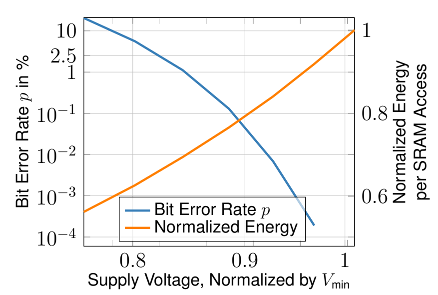


Bit error **robustness**:

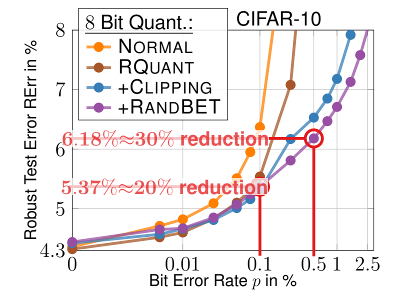


1-Minute Overview: Bit Error Robustness

Random **bit errors**:



Bit error **robustness**:



More details:

Paper & code: `davidstutz.de/randbet`

Contact: `david.stutz@mpi-inf.mpg.de`

Interested?

More details:

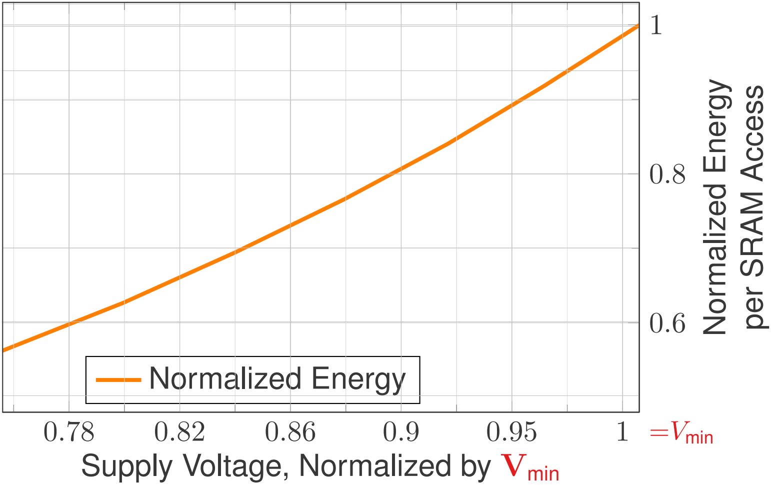
Paper & code: `davidstutz.de/randbet`

Contact: `david.stutz@mpi-inf.mpg.de`

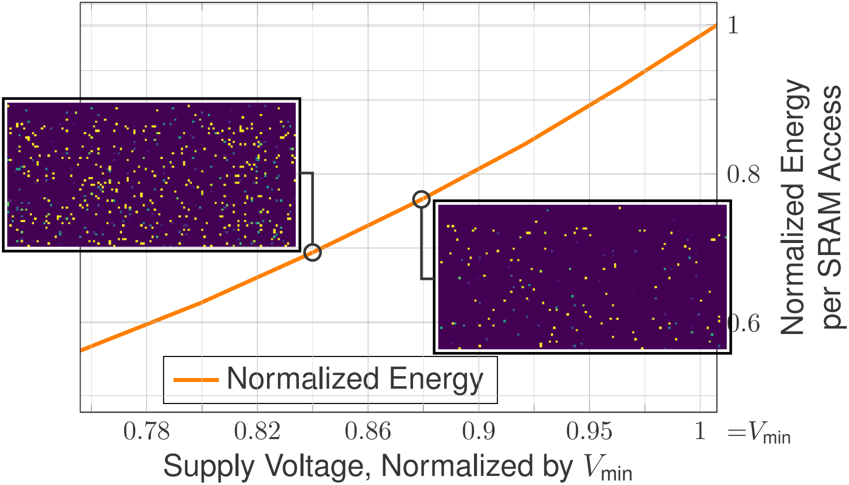
Outline:

1. Bit errors in DNN accelerators
2. Error model and contributions
3. Robust quantization, weight clipping, and random bit error training
4. Results and energy savings

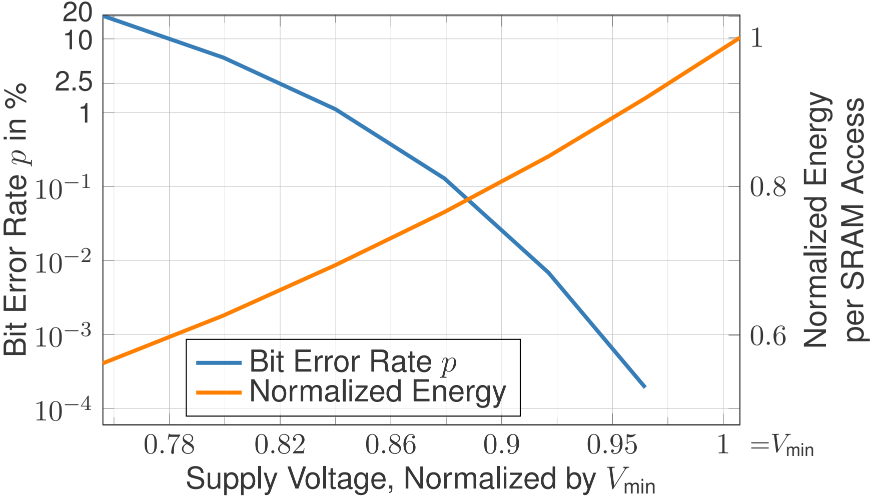
Energy Consumption in DNN Accelerators



Energy Consumption in DNN Accelerators

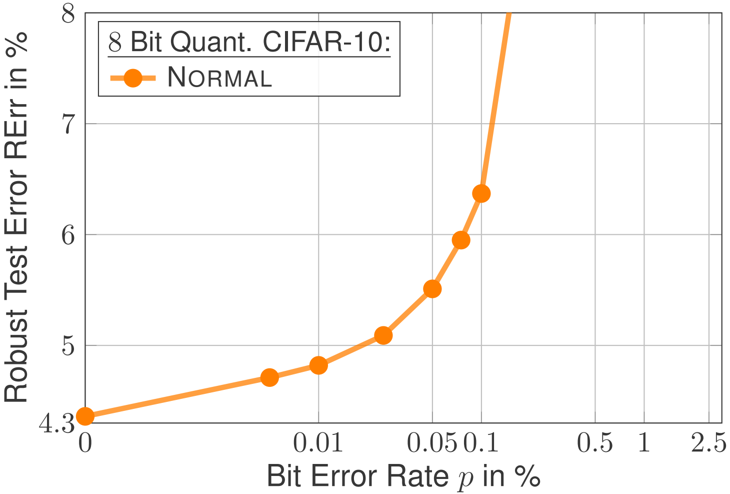


Low-Voltage Operation and Bit Errors



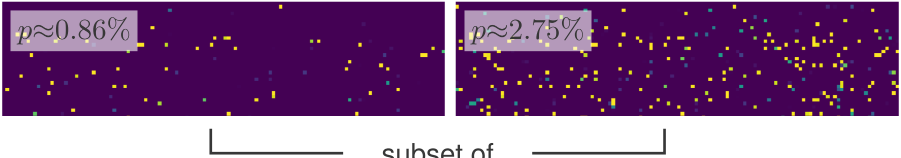
Bit Error Impact on DNNs on CIFAR-10

Axis Changed!



Bit Error Model and Contributions

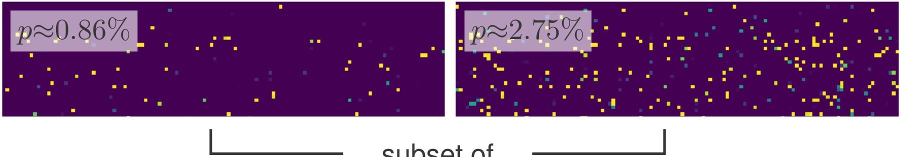
<u>Bit error model:</u>



- Uniform (across locations+chips) random bit errors.

Bit Error Model and Contributions

Bit error model:



- Uniform (across locations+chips) random bit errors.

Contributions:

- Robust fixed-point quantization (RQUANT).
- Weight clipping as regularization (CLIPPING).
- Random bit error training (RANDBET).

Robust Quantization (RQUANT)

Simple fixed-point quantization scheme:

$$Q(w_i) = \left\lfloor \frac{w_i}{\Delta} \right\rceil, Q^{-1}(v_i) = \Delta v_i, \Delta = \frac{q_{\text{max}}}{w^{m-1} - 1}$$

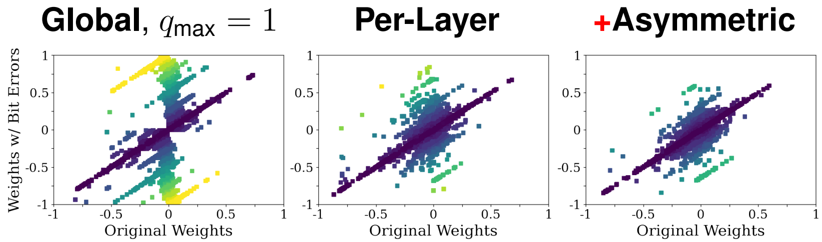
- weight $w_i \in [-q_{\text{max}}, q_{\text{max}}]$, m bits (e.g., $m = 8$)

Robust Quantization (RQuant)

Simple fixed-point quantization scheme:

$$Q(w_i) = \left\lfloor \frac{w_i}{\Delta} \right\rceil, Q^{-1}(v_i) = \Delta v_i, \Delta = \frac{q_{\text{max}}}{w^{m-1} - 1}$$

- weight $w_i \in [-q_{\text{max}}, q_{\text{max}}]$, m bits (e.g., $m = 8$)



Robust Quantization (RQuant)

Simple fixed-point quantization scheme:

$$Q(w_i) = \left\lfloor \frac{w_i}{\Delta} \right\rfloor, Q^{-1}(v_i) = \Delta v_i, \Delta = \frac{q_{\text{max}}}{w^{m-1} - 1}$$

Importance of **implementation details**:

	Quantization Scheme (CIFAR-10, BER $p = 0.5\%$)	Err in %	RErr in %
8 bit	Per-layer	4.36	24.76
	+asymmetric	4.36	40.78
	+unsigned	4.42	17.00
	+rounding (=RQuant)	4.32	**11.28**

Robust Quantization (RQUANT)

Simple fixed-point quantization scheme:

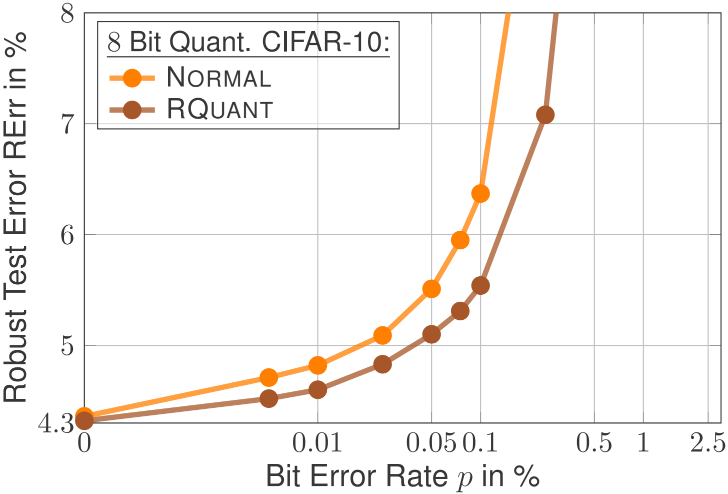
$$Q(w_i) = \left\lfloor \frac{w_i}{\Delta} \right\rceil, Q^{-1}(v_i) = \Delta v_i, \Delta = \frac{q_{\text{max}}}{w^{m-1} - 1}$$

Importance of **implementation details**:

	Quantization Scheme (CIFAR-10, BER $p = 0.5\%$)	Err in %	RErr in %
4 bit	w/o rounding*	5.81	90.36
4 bit	w/ rounding*	**5.29**	**7.71**

*Results with weight clipping.

Robust Quantization (RQuant)



Weight Clipping as Regularization (CLIPPING)

= clipping weights to $[-w_{\text{max}}, w_{\text{max}}]$ during training.

Important:

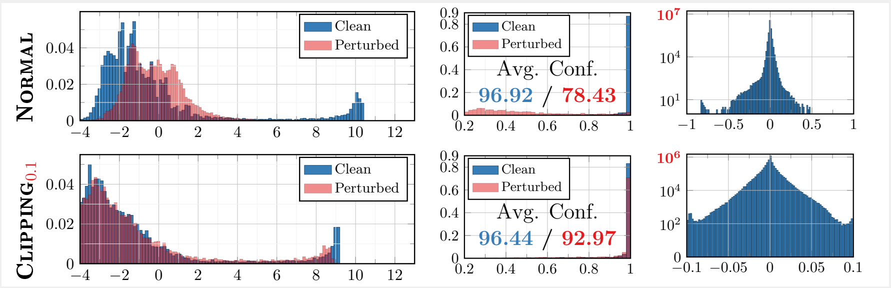
- $w_{\text{max}} \neq q_{\text{max}}$, but $q_{\text{max}} \leq w_{\text{max}}$
- Does *not* impact *relative* errors!

Weight Clipping as Regularization (Clipping)

= clipping weights to $[-q_{\max}, w_{\max}]$ during training.

Important:

- $w_{\max} \neq q_{\max}$, but $q_{\max} \leq w_{\max}$
- Does *not* impact *relative* errors!



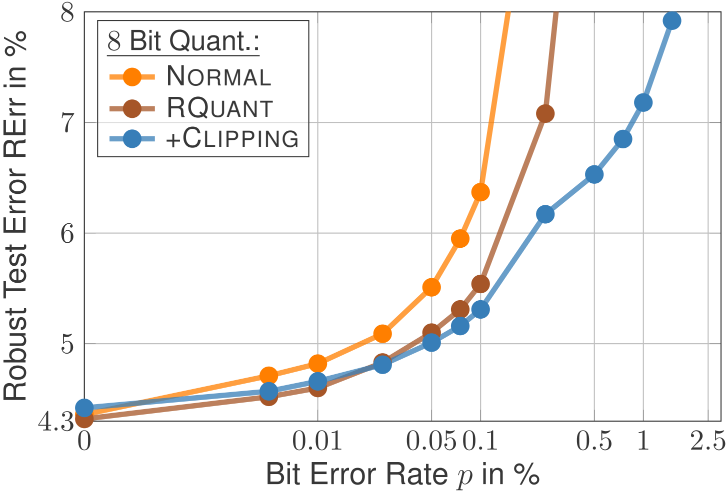
Weight Clipping as Regularization (CLIPPING)

Why does CLIPPING improve bit error robustness?

- Limiting weights and minimizing cross-entropy loss
- Large logits achievable through weight redundancy

Model (CIFAR-10, BER $p{=}1\%$)	Err in %	RErr in %
RQUANT	**4.32**	32.05
CLIPPING$_{0.15}$	4.42	13.08
CLIPPING$_{0.15}$+label smoothing	4.67	29.40

Weight Clipping as Regularization (Clipping)



Random bit error training (RANDBET)

= training on *random* bit errors

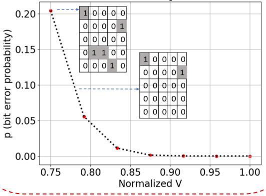
$w^{(t)}$

$v^{(t)} = Q\left(w^{(t)}\right)$

Quantize

$\tilde{v}^{(t)} = BErr_p\left(v^{(t)}\right)$

Inject errors



$w_q^{(t)} = Q^{-1}\left(v^{(t)}\right)$

$\tilde{w}_q^{(t)} = Q^{-1}\left(\tilde{v}^{(t)}\right)$

Dequantize

$$\Delta^{(t)} = \nabla_w \sum_b L\left(f\left(x_b, w_q^{(t)}\right), y_b\right)$$

$$\tilde{\Delta}^{(t)} = \nabla_w \sum_b L\left(f\left(x_b, \tilde{w}_q^{(t)}\right), y_b\right)$$

Forward and backward pass for unperturbed & perturbed models

$$w^{(t+1)} = w^{(t)} - \gamma\left(\Delta^{(t)} + \tilde{\Delta}^{(t)}\right)$$

weight update with average of gradients

Random bit error training (RandBET)

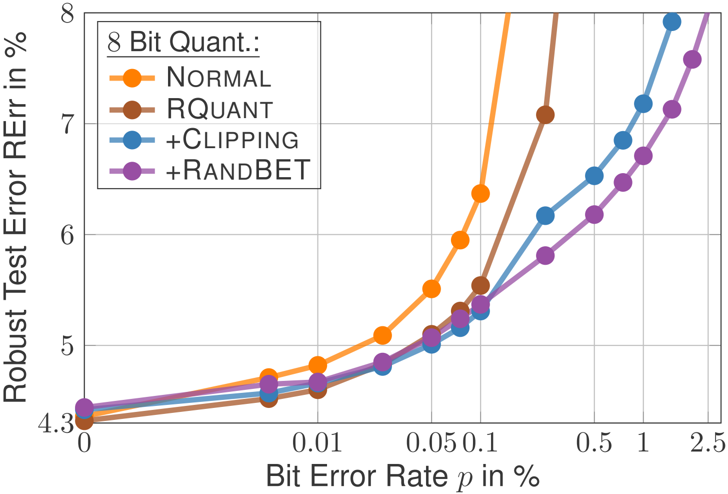
Important: train on *random* bit errors.

Related work frequently trains on *profiled* bit errors.

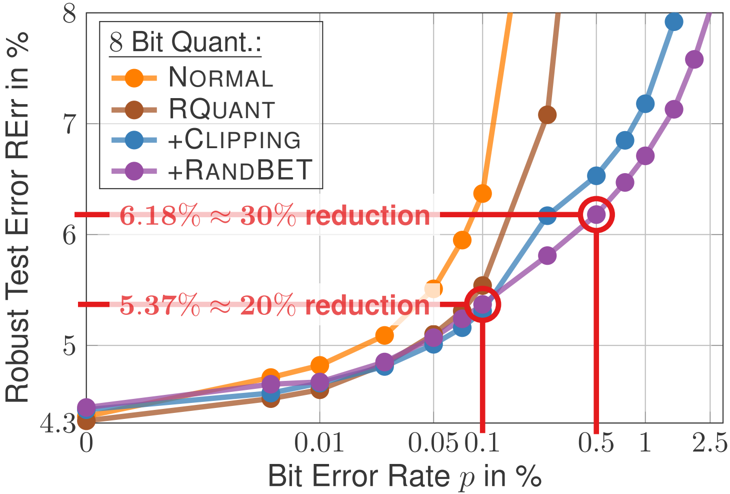
(Specific to *one* chip *and* voltage.)

Model	RErr in %, p in %	
Evaluation on Fixed Pattern	p=1	p=2.5
Fixed Pattern p=2.5	14.14	7.87
Fixed Pattern+Clipping$_{0.15}$ p=2.5	8.50	7.41
Evaluation on *Random* Patterns	p=1	p=2.5
Fixed Pattern+Clipping$_{0.15}$ p=2.5	12.09	61.59

Random bit error training (RandBET)

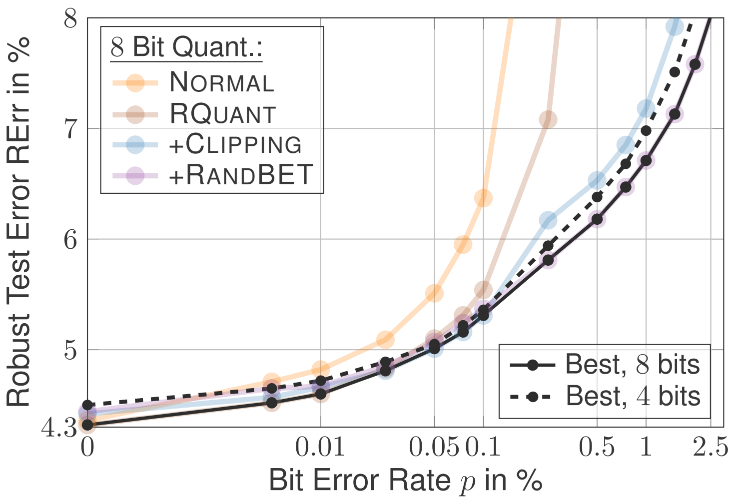


Random bit error training (RANDBET)

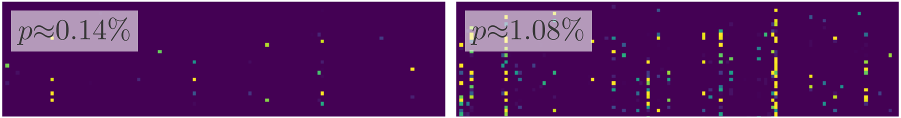


max planck institut informatik
EBERHARD KARLS UNIVERSITÄT TÜBINGEN

Low-Voltage *and* Low-Precision



Generalization Across Chips/Voltages

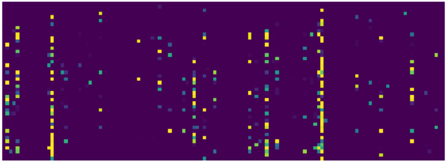


subset of

- ► "Corner-cases" might exhibit different error patterns.

Chip	Model (CIFAR-10)	RErr in %	
Chip 1		$p{\approx}0.86$	$p{\approx}2.75$
	RandBET$_{0.05}$ $p{=}1.5$	7.04	9.37
Chip 2	(see above)	$p{\approx}0.14$	$p{\approx}1.08$
	RandBET$_{0.05}$ $p{=}1.5$	6.00	9.00

Bit Error Robustness for DNN Accelerators



Conclusion:

- Uniform bit error model.
- Robust quantization.
- Weight clipping and random bit error training.
- Generalization across chips and voltages.

Paper: `https://davidstutz.de/randbet`

- Results on MNIST / CIFAR-100, guarantees, ...