Centered Symmetric Quantization for Hardware-Efficient Low-bit Neural Networks

ULSAN NATIONAL INSTITUTE OF SCIENCE AND TECHNOLOGY

Faaiz Asim, Jaewoo Park*, Azat Azamat, Jongeun Lee*

BMVC 2022

Introduction

Problem:

- **Conventional Linear Quantization (CLQ)** results in unequal number of positive and negative quantization levels (e.g 2-bit signed CLQ has {-2, -1, 0, 1} quantization levels)
- A perfectly symmetric linear quantizer can not be realized using standard multipliers without large overhead

Contributions:

- We propose **Centered Symmetric Quantization (CSQ)**
- We propose efficient methods to realize **CSQ** on hardware:
  1. Using standard multiliers with small overhead
  2. Using Binarized Neural Network (BNN) hardware with no overhead

CSQ Definition

We define Centered Symmetric Quantization (CSQ) as a quantizer that has:

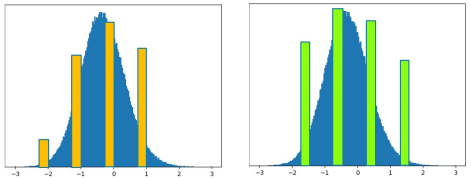
1. Uniform step-size between quantization levels
2. Perfectly symmetric quantization levels

CSQ is formulated as:

$$\dot{v} = \left\lfloor \frac{v}{s} + 0.5 \right\rceil - 0.5$$

$$\bar{v} = \text{clip}\,(\dot{v}, -Q, Q)$$

$$\hat{v} = \bar{v} \times s$$



Conventional Linear Quantizer (CLQ)

Centered Symmetric Quantizer (CSQ)

CSQ Hardware Realization

Method 1: On standard multipliers using affine quantization

Zero-point for **CSQ** and **CLQ** in context of affine quantization:

$$z_{CLQ} = 2^{b-1}\,,\; z_{CSQ} = 2^{b-1} - 0.5$$

$$\therefore z_{CSQ} = z_{CLQ} - 0.5$$

Realizing **CSQ** as affine quantizer: $\hat{w}\hat{x} = ((\bar{w} - 0.5) \times s_w)\,(\bar{x} \times s_x) = \bar{w}\bar{x}s_w s_x - \underbrace{0.5 \times s_w s_x \bar{x}}_{\text{overhead}}$

+ Significantly smaller overhead compared to affine quantizer
+ Flexible method generalizeable to any hardware
- Results in small overhead

2-bit CSQ Integer	CSQ binary encoding	
	2^1	2^0
-3	0	0
-1	0	1
1	1	0
3	1	1

2-bit CSQ binary encoding

Method 2: Bitwise BNN-hardware based method

An *n*-bit **CSQ** number can be represented in binary using +1 and -1, instead of 0 and 1. (See 2-bit **CSQ** binary encoding example)
For signed weights and unsigned activations , **CSQ-CLQ** multiplication can be computed using AND-popcount:

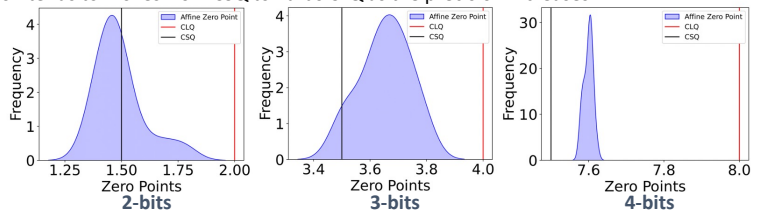
$$v_{csq} \cdot x_{clq_u} = 2 \cdot popcount\left(AND\left(v_{csq}, x_{clq_u}\right)\right) - popcount(x_{clq_u})$$

+ No overhead
+ Configurable precision due to bit-wise operations
- Cannot be realized using standard multipliers

Analyzing CSQ vs CLQ

Learned per-layer zero-point distribution for ResNet-20 compared to CSQ vs. CLQ

- Zero-point distribution is closer to CSQ than CLQ especially at 2-bit precision
- Zero-point distribution tends to moves from CSQ towards CLQ as the precision increases



ImageNet Results

Network	Top-1 Accuracy @ Precision								
	ResNet-18			ResNet-34			MobileNet-v2		
	Full Precision: 70.58			*Full Precision: 73.31*			*Full Precision: 71.88*		
Precision (W/A)	2/2	3/3	4/4	2/2	3/3	4/4	2/2	3/3	4/4
PACT	64.40	68.10	69.20	-	-	-	-	-	-
LQ-Nets	64.90	68.20	69.30	69.80	71.90	-	-	-	-
QIL	65.70	69.20	70.10	70.60	73.10	73.70	-	-	-
CLQ (LSQ)	66.59	69.38	70.52	70.56	73.21	73.82	-	60.41	66.82
CSQ (LSQ)	**66.92**	**69.48**	**70.63**	**70.82**	**73.29**	**74.01**	-	**60.89**	**66.98**

FPGA Results

Precision	2-bit				3-bit			
	Utilization		Latency	ADP	Utilization		Latency	ADP
	LUT	FF	(ns)	(%)	LUT	FF	(ns)	(%)
CLQ w/ Mult.	897	671	2.07	100	1171	1009	2.11	100
CLQ w/ BNN HW	673	624	2.20	88.8	992	926	2.31	96.5
CSQ w/ BNN HW	681	630	2.24	91.4	1001	944	2.35	99.6
(Relaxed) Affine Quantizer	1002	849	2.81	161	1362	1277	2.89	166
CSQ w/ Affine HW	913	680	2.33	114	1207	1015	2.42	117

Conclusion

- **CSQ** for weight quantization can provide significant performance improvement compared to **CLQ** at extremely low precision (≤3-bits)
- At higher precisions (≥4-bits) the performance improvement using **CSQ** diminishes.
- For standard multipliers (e.g., GPU, CPU), our **affine quantization-based** hardware realization method allows **CSQ** realization with a very small overhead.
- Our **bitwise BNN hardware-based** method allows **CSQ** realization without any overhead.