CROQuant: Complex Rank-One Quantization Algorithm

Maël Chaumette Rémi Gribonval Elisa Riccietti

OCKHAM Team

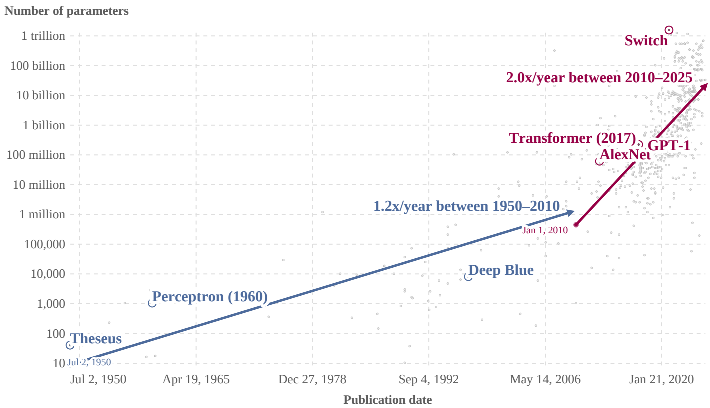
RAIM Meeting 2025

6/11/2025

ENS DE LYON

RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

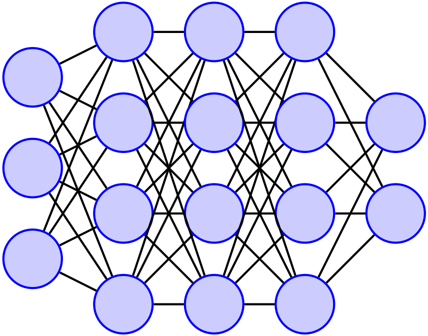
Context



Explosion in the size of deep learning models.
Source: [Samborska, 2025]

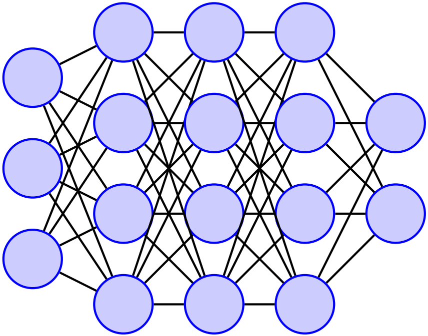
Objective

- **Main goal:** Quantize the weights of a neural network



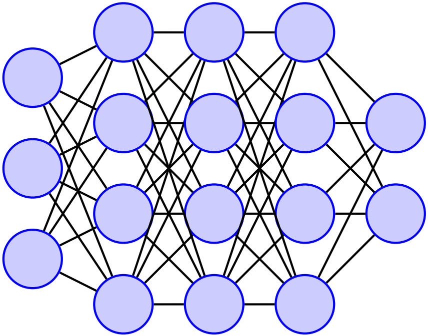
Objective

- **Main goal:** Quantize the weights of a neural network
- **Naive approach:** Map each weight to its nearest neighbor in the quantization set



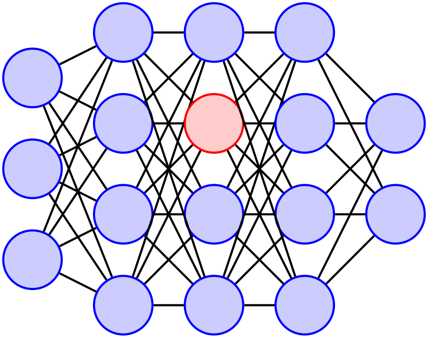
Objective

- **Main goal:** Quantize the weights of a neural network
- **Naive approach:** Map each weight to its nearest neighbor in the quantization set
- **Opportunity:** Take into account the *rescaling invariance* property of ReLU neural networks [Neyshabur et al., 2015]



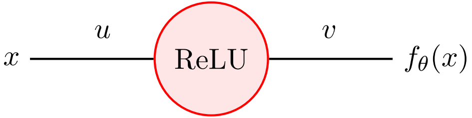
Objective

- **Main goal:** Quantize the weights of a neural network
- **Naive approach:** Map each weight to its nearest neighbor in the quantization set
- **Opportunity:** Take into account the *rescaling invariance* property of ReLU neural networks [Neyshabur et al., 2015]



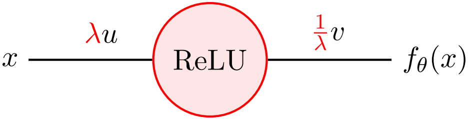
What is the rescaling invariance property?

- $\theta = (u, v) \in \mathbb{R}^{m+n}$
- $f_\theta : x \in \mathbb{R}^m \mapsto \mathrm{ReLU}(\langle u, x \rangle)v \in \mathbb{R}^n$

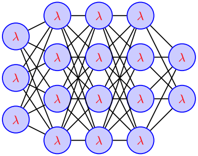


What is the rescaling invariance property?

- $\theta = (u, v) \in \mathbb{R}^{m+n}$
- $f_\theta : x \in \mathbb{R}^m \mapsto \mathrm{ReLU}(\langle \lambda u, x \rangle) \frac{1}{\lambda} v \in \mathbb{R}^n$ with $\lambda > 0$

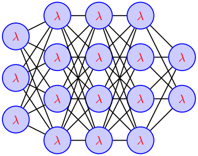


Outline



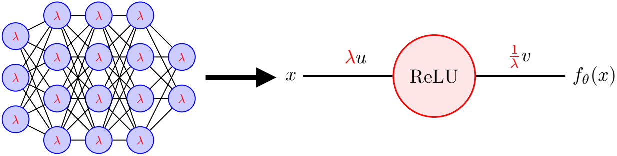
Outline

❌ too many λ's



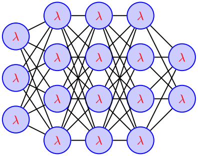
Outline

❌ too many λ's

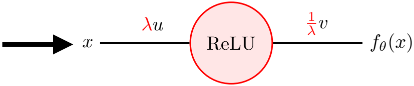


Outline

❌ too many λ's

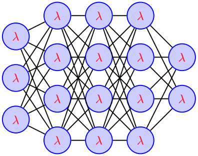


❌ non-linearity

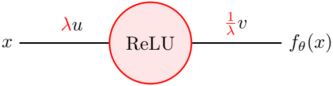


Outline

too many λ's

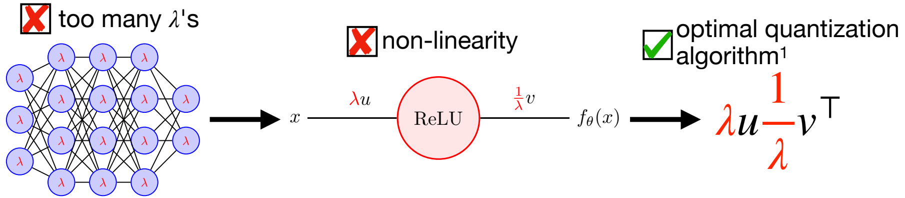


non-linearity



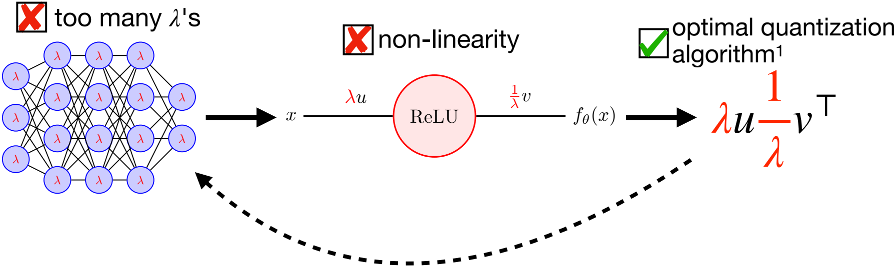
$$\lambda u \frac{1}{\lambda} v^\top$$

Outline



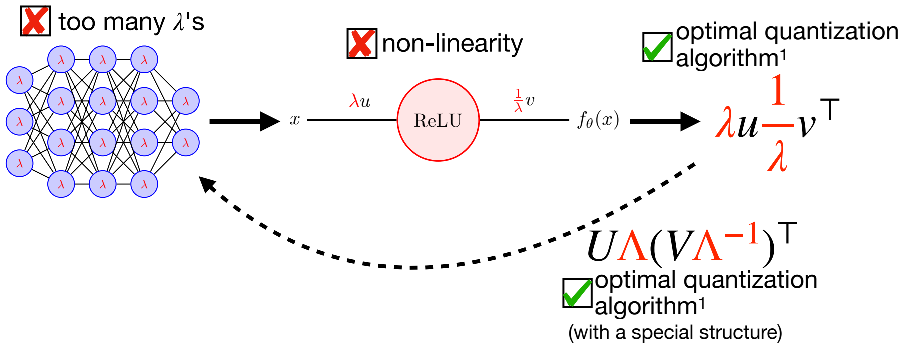
[1] Proved by Gribonval et al. [2023]

Outline



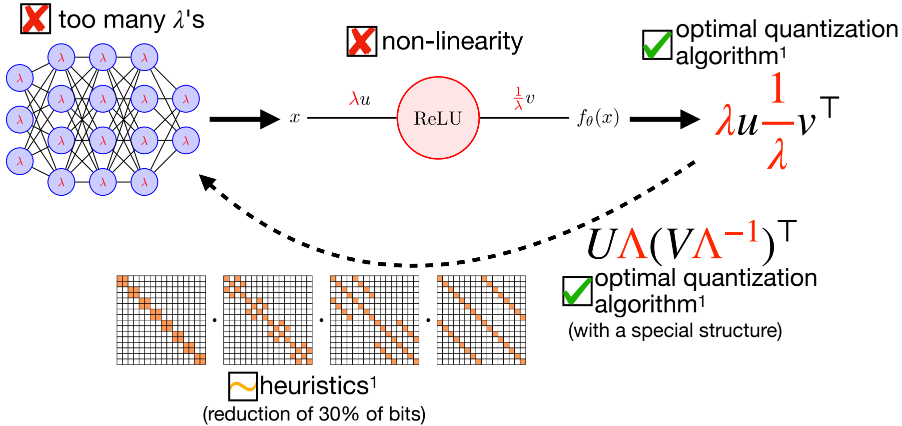
[1] Proved by Gribonval et al. [2023]

Outline



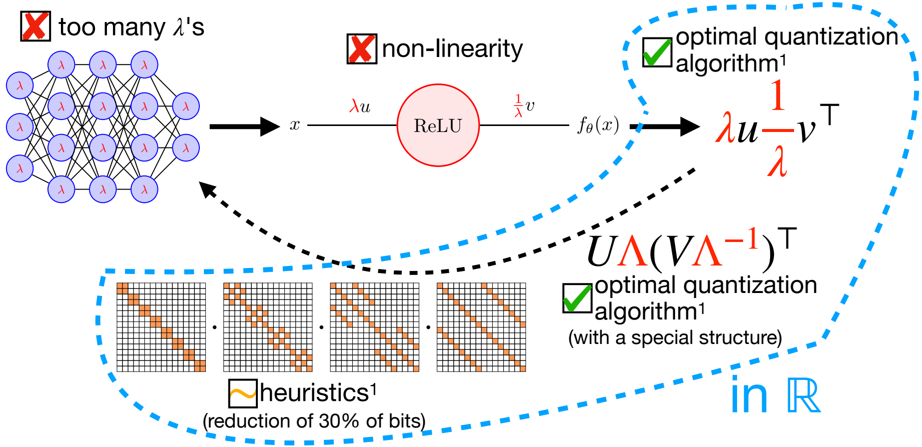
[1] Proved by Gribonval et al. [2023]

Outline



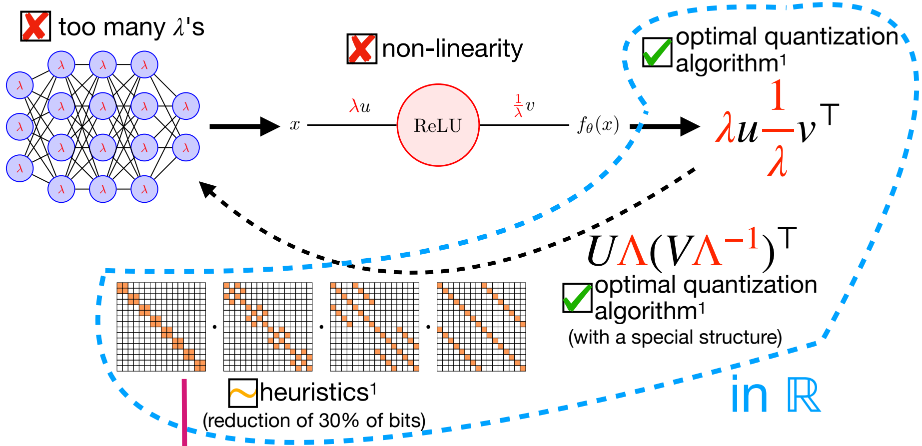
[1] Proved by Gribonval et al. [2023]

Outline



[1]Proved by Gribonval et al. [2023]

Outline



Appear in the FFT $\longrightarrow$ extension to $\mathbb{C}$?

[1] Proved by Gribonval et al. [2023]

Quantization of complex rank-one matrices

$$\forall \lambda \in \mathbb{C}^*,\ \ (\lambda x) \left(\frac{1}{\bar{\lambda}} y\right)^H$$

Problem statement

- $\mathbb{F}_t$: set of floating-point numbers with t-bit significand
- $\mathbb{CF}_t := \mathbb{F}_t + i\mathbb{F}_t$

Problem statement

- $\mathbb{F}_t$: set of floating-point numbers with t-bit significand
- $\mathbb{CF}_t := \mathbb{F}_t + i\mathbb{F}_t$

Problem formulation

Given $(x, y) \in \mathbb{C}^m \times \mathbb{C}^n$, we want to solve:

$$\min_{\hat{x} \in \mathbb{CF}_t^m, \hat{y} \in \mathbb{CF}_t^n} \|xy^H - \hat{x}\hat{y}^H\|^2$$

Problem statement

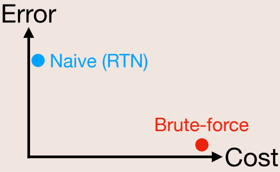
- $\mathbb{F}_t$: set of floating-point numbers with t-bit significand
- $\mathbb{CF}_t := \mathbb{F}_t + i\mathbb{F}_t$

Problem formulation

Given $(x, y) \in \mathbb{C}^m \times \mathbb{C}^n$, we want to solve:

$$\min_{\hat{x} \in \mathbb{CF}_t^m, \hat{y} \in \mathbb{CF}_t^n} \|xy^H - \hat{x}\hat{y}^H\|^2$$

Potential approaches



Problem statement

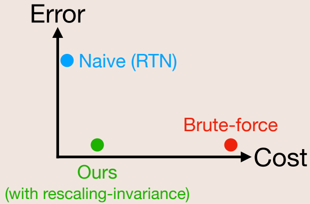
- $\mathbb{F}_t$: set of floating-point numbers with t-bit significand
- $\mathbb{CF}_t := \mathbb{F}_t + i\mathbb{F}_t$

Problem formulation

Given $(x, y) \in \mathbb{C}^m \times \mathbb{C}^n$, we want to solve:

$$\min_{\hat{x} \in \mathbb{CF}_t^m, \hat{y} \in \mathbb{CF}_t^n} \|xy^H - \hat{x}\hat{y}^H\|^2$$

Potential approaches



Problem resolution

Key (simple) lemma: problem characterization (reminder of the real-case)

Given $(x, y) \in \mathbb{C}^m \times \mathbb{C}^n$, there exists a function $f : \mathbb{R} \mapsto \mathbb{R}_+$ such that

$$\hat{x}^* = \mathrm{round}(\lambda^* x) \text{ and } \hat{y}^* = \mathrm{round}(\mu^*(\lambda^*) y)$$

$$\text{where } \lambda^* \in \arg\min_{\mathbb{R}} f$$

Problem resolution

Key (simple) lemma: problem characterization (in the complex-case)

Given $(x, y) \in \mathbb{C}^m \times \mathbb{C}^n$, there exists a function $f : \mathbb{C} \mapsto \mathbb{R}_+$ such that

$$\hat{x}^* = \mathrm{round}(\lambda^* x) \text{ and } \hat{y}^* = \mathrm{round}(\mu^*(\lambda^*) y)$$

$$\text{where } \lambda^* \in \arg\min_{\mathbb{C}} f$$

Problem resolution

Key (simple) lemma: problem characterization (in the complex-case)

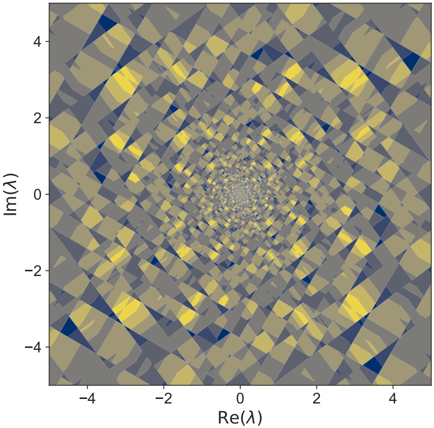
Given $(x, y) \in \mathbb{C}^m \times \mathbb{C}^n$, there exists a function $f : \mathbb{C} \mapsto \mathbb{R}_+$ such that

$$\hat{x}^* = \mathrm{round}(\lambda^* x) \text{ and } \hat{y}^* = \mathrm{round}(\mu^*(\lambda^*) y)$$

$$\text{where } \lambda^* \in \arg\min_{\mathbb{C}} f$$

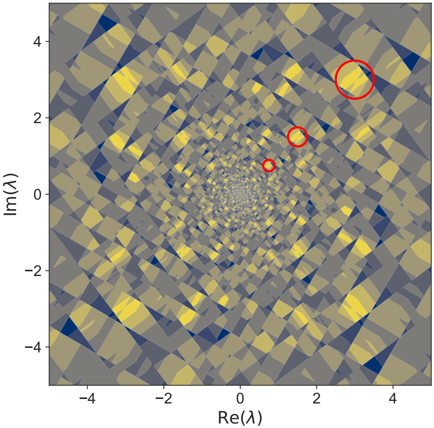
$\rightarrow$ Reduction of a problem with $2(m+n)$ variables to a one scalar problem.

Study of the function f



Properties of f:

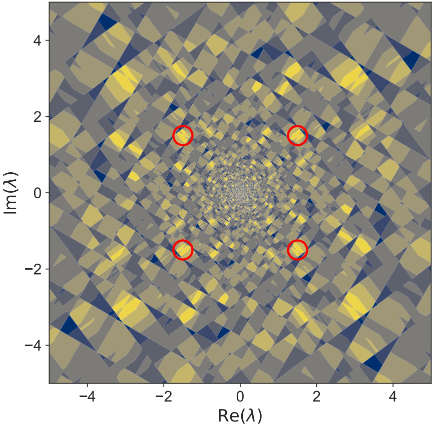
Study of the function f



Properties of f:

- f is **invariant** by multiplication by 2

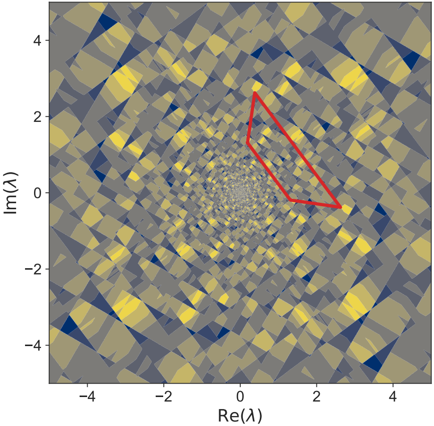
Study of the function f



Properties of f:

- f is **invariant** by multiplication by 2 and i

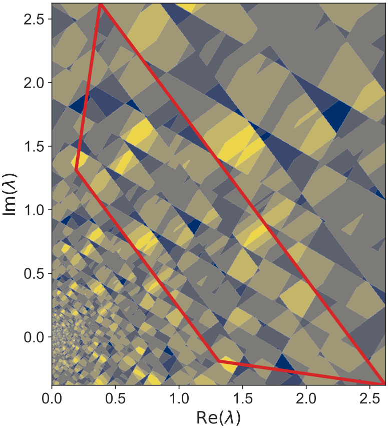
Study of the function f



Properties of f:

- f is **invariant** by multiplication by 2 and i

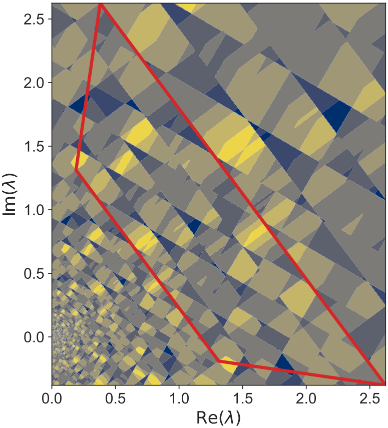
Study of the function f



Properties of f:

- f is **invariant** by multiplication by 2 and i

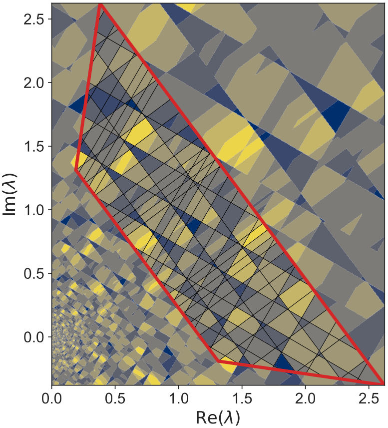
Study of the function f



Properties of f:

- f is **invariant** by multiplication by 2 and i
- f is piecewise constant

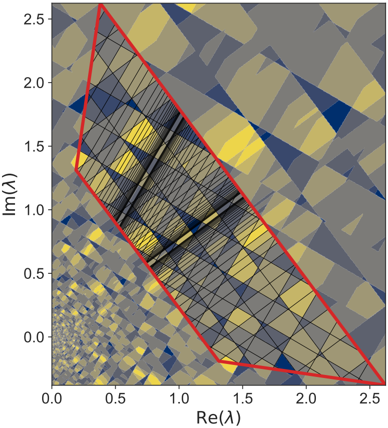
Study of the function f



Properties of f:

- f is **invariant** by multiplication by 2 and i
- f is piecewise constant where discontinuity points are **lines**

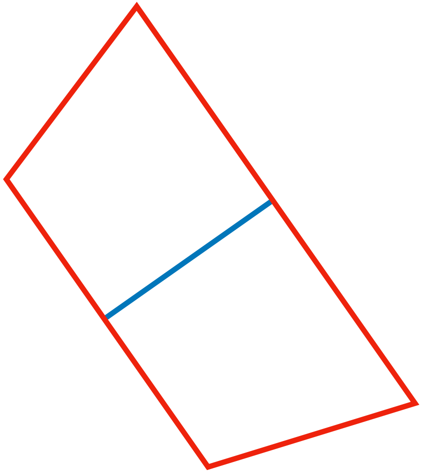
Study of the function f



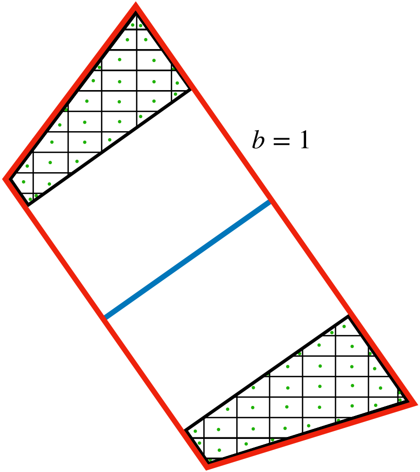
Properties of f:

- f is **invariant** by multiplication by 2 and i
- f is piecewise constant where discontinuity points are **lines**
- **But** with an infinite number of lines

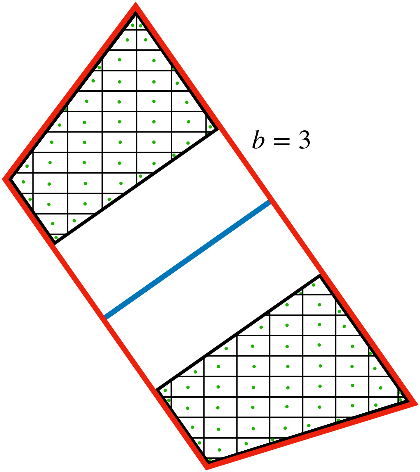
Definition of the algorithm



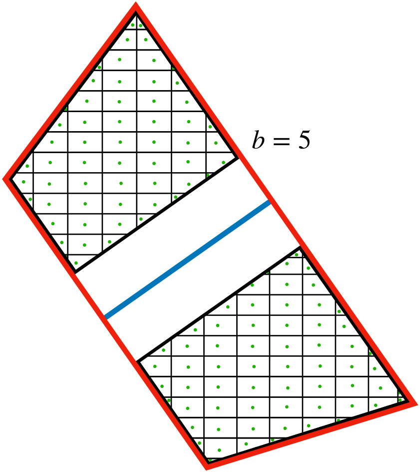
Definition of the algorithm



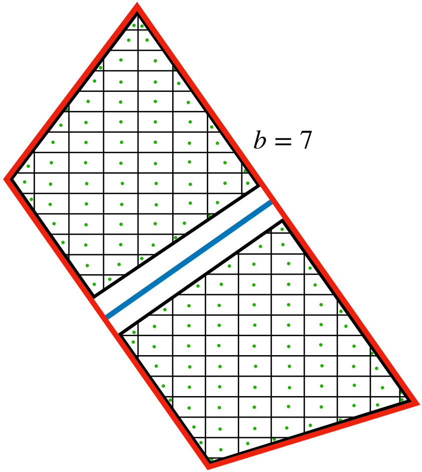
Definition of the algorithm



Definition of the algorithm



Definition of the algorithm

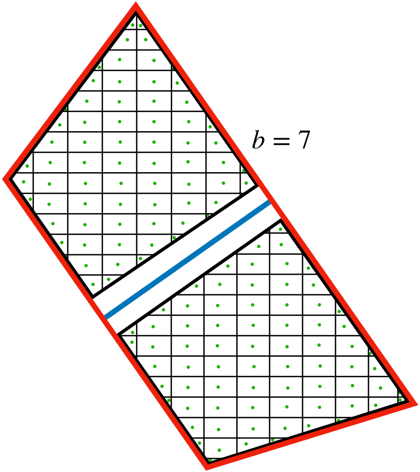


Algorithm steps

Iterate until b is reached:

1. compute all the centroids from the polygonal pieces
2. evaluate f on all these centroids
3. keep the **best** scaling factor, λ_b
4. return $\hat{x}_b := \mathrm{round}(\lambda_b x)$ and $\hat{y}_b := \mathrm{round}(\mu_b y)$

Definition of the algorithm



Algorithm steps

Iterate until b is reached:

1. compute all the centroids from the polygonal pieces
2. evaluate f on all these centroids
3. keep the **best** scaling factor, λ_b
4. return $\hat{x}_b := \mathrm{round}(\lambda_b x)$ and $\hat{y}_b := \mathrm{round}(\mu_b y)$

$\rightarrow$ **Lemma:** there exists $\tilde{b} < +\infty$ that finds the optimal solution

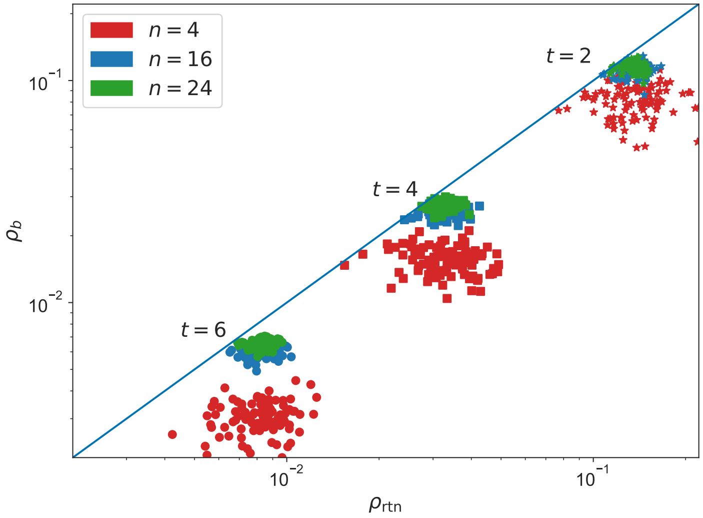
Comparison with the baseline

Metric definition:

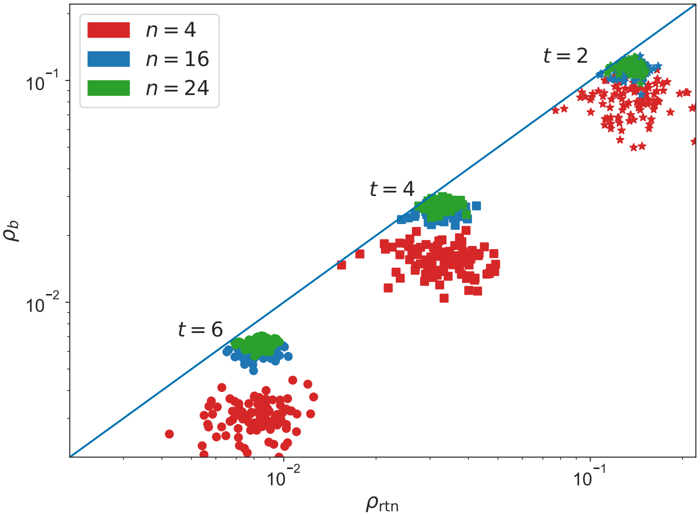
$$\rho(\hat{x}, \hat{y}) := \|xy^H - \hat{x}\hat{y}^H\| / \|xy^H\|$$

$$\rho_b := \rho(\hat{x}_b, \hat{y}_b) \text{ and } \rho_{\mathsf{rtn}} := \rho(\mathrm{round}(x), \mathrm{round}(y))$$

Comparison with the baseline $(b = 3)$

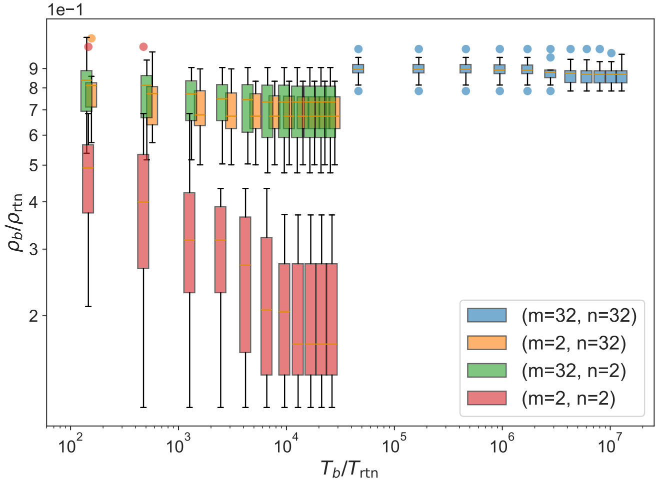


Comparison with the baseline $(b = 3)$

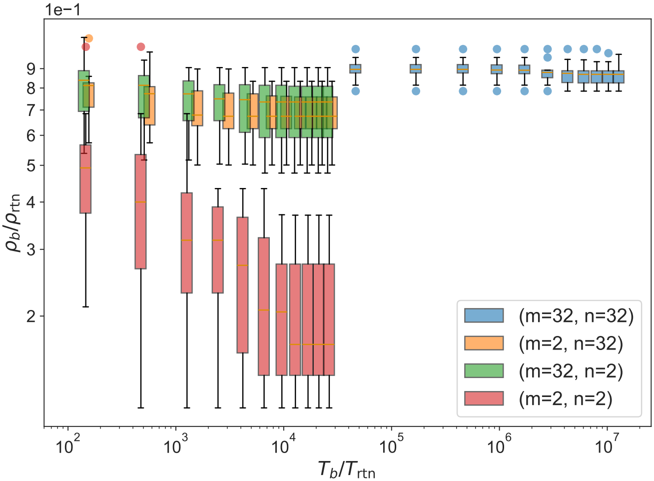


$\rightarrow$ Our algorithm is more accurate than the naive rounding approach

Role of the dimension and the parameter b $(t = 4)$



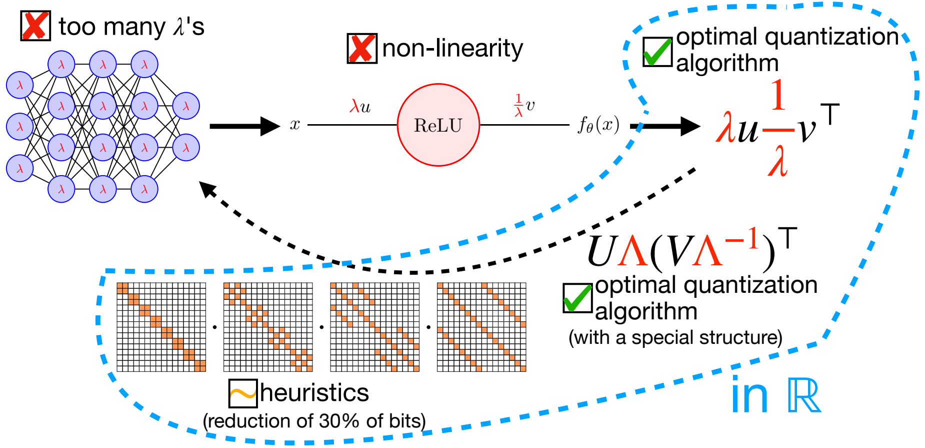
Role of the dimension and the parameter b $(t = 4)$



$\rightarrow$ Our algorithm is more interesting for small vectors

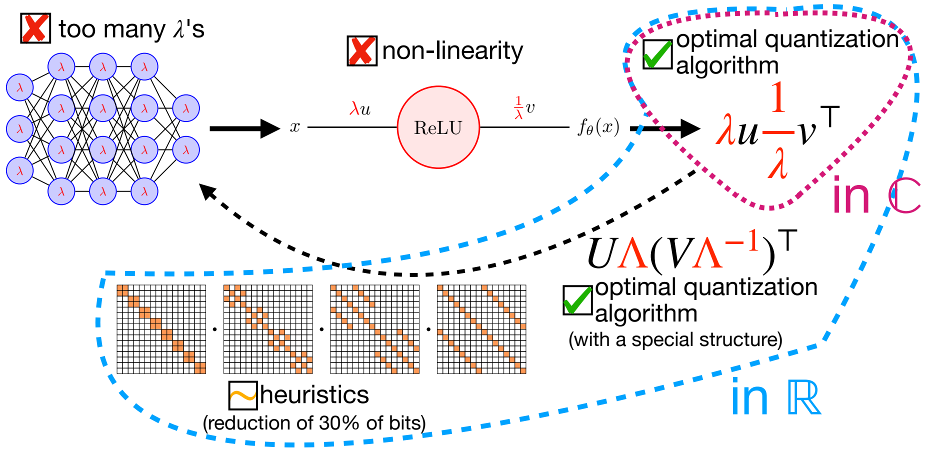
In this case, increasing b improves significantly the accuracy

Outline



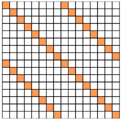
too many λ's
non-linearity
optimal quantization algorithm
x
λu
ReLU
$\frac{1}{\lambda} v$
$f_\theta(x)$
$\lambda u \frac{1}{\lambda} v^\top$
$U\Lambda(V\Lambda^{-1})^\top$
optimal quantization algorithm
(with a special structure)
heuristics
(reduction of 30% of bits)
in ℝ

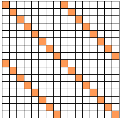
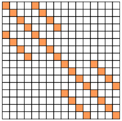
Outline


too many λ's
non-linearity
x
λu
ReLU
$\frac{1}{\lambda} v$
$f_\theta(x)$
optimal quantization algorithm
$\lambda u \frac{1}{\lambda} v^\top$
in ℂ
$U\Lambda(V\Lambda^{-1})^\top$
optimal quantization algorithm
(with a special structure)
heuristics
(reduction of 30% of bits)
in ℝ

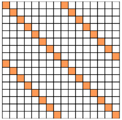
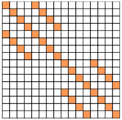
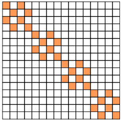
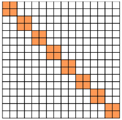
Application to butterfly matrices

What's FFT?

$F_n =$  $\begin{pmatrix} F_{n/2} & 0 \\ 0 & F_{n/2} \end{pmatrix}$ (Sort the even and odd indices)

$=$   $\begin{pmatrix} F_{n/4} & 0 & 0 & 0 \\ 0 & F_{n/4} & 0 & 0 \\ 0 & 0 & F_{n/4} & 0 \\ 0 & 0 & 0 & F_{n/4} \end{pmatrix}$ (Permutation)

$\vdots$

$=$     (Permutation)

$\underbrace{B_1 \quad B_2 \quad B_3 \quad B_4}$

$L := \log_2(n)$ butterfly factors

[Cooley and Tukey, 1965]

Complex butterfly quantization

Objective

Consider $B_1, ..., B_L \in \mathbb{C}^{n\times n}$. We want to solve

$$B_1^*, ..., B_L^* \in \arg\min_{\hat{B}_1,...,\hat{B}_L \in \mathbb{C}^{n\times n}} \|B_1 \cdots B_L - \hat{B}_1 \cdots \hat{B}_L\|^2$$

Complex butterfly quantization

Objective

Consider $B_1, ..., B_L \in \mathbb{C}^{n\times n}$. We want to solve

$$B_1^*, ..., B_L^* \in \arg \min_{\hat{B}_1,...,\hat{B}_L \in \mathbb{C}^{n\times n}} \|B_1 \cdots B_L - \hat{B}_1 \cdots \hat{B}_L\|^2$$

- For $L = 2$: **Solvable** problem because it can be written as n independent **rank-one quantization problems**
- For $L > 2$: use **parentheses** to express subproblems with $L = 2$

Complex butterfly quantization

Objective

Consider $B_1, ..., B_L \in \mathbb{C}^{n \times n}$. We want to solve

$$B_1^*, ..., B_L^* \in \arg \min_{\hat{B}_1, ..., \hat{B}_L \in \mathbb{C}^{n \times n}} \| B_1 \cdots B_L - \hat{B}_1 \cdots \hat{B}_L \|^2$$

- For $L = 2$: **Solvable** problem because it can be written as n independent **rank-one quantization problems**
- For $L > 2$: use **parentheses** to express subproblems with $L = 2$

Heuristics for the parenthesis decomposition

Pairwise: writing $(B_1 B_2)(B_3 B_4) \cdots (B_{L-1} B_L)$
Left-to-Right (LTR): writing $B_1(B_2(\cdots(B_{L-1} B_L)))$

Complex butterfly quantization

Objective

Consider $B_1, ..., B_L \in \mathbb{C}^{n \times n}$. We want to solve

$$B_1^*, ..., B_L^* \in \arg \min_{\hat{B}_1, ..., \hat{B}_L \in \mathbb{C}^{n \times n}} \|B_1 \cdots B_L - \hat{B}_1 \cdots \hat{B}_L\|^2$$

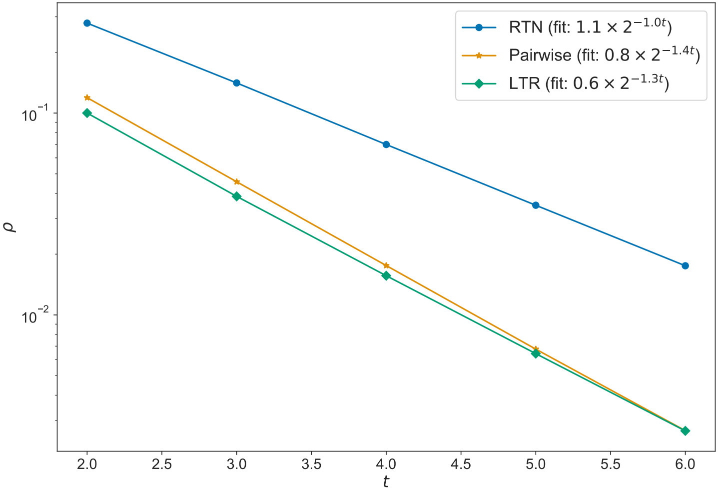
- For $L = 2$: **Solvable** problem because it can be written as n independent **rank-one quantization problems**
- For $L > 2$: use **parentheses** to express subproblems with $L = 2$

Heuristics for the parenthesis decomposition

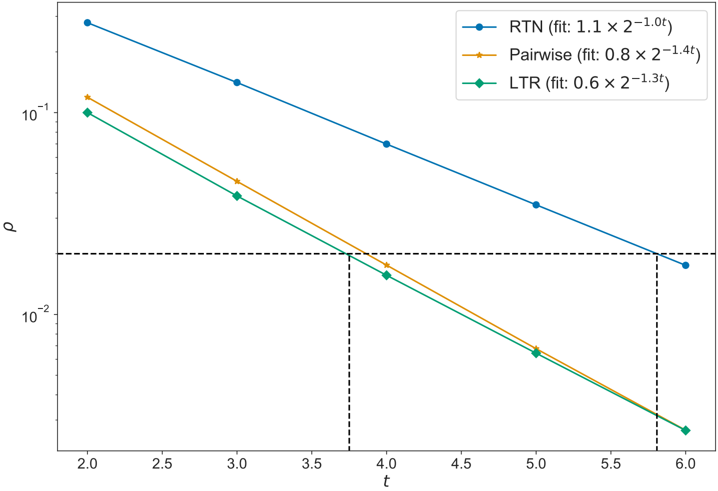
Pairwise: writing $(B_1 B_2)(B_3 B_4) \cdots (B_{L-1} B_L)$
Left-to-Right (LTR): writing $B_1(B_2(\cdots(B_{L-1} B_L)))$

The metric is $\rho := \dfrac{\|B_1 \cdots B_L - \hat{B}_1 \cdots \hat{B}_L\|}{\|B_1 \cdots B_L\|}$

Comparison with the baseline in terms of t ($n = 256$ and $b = 3$)

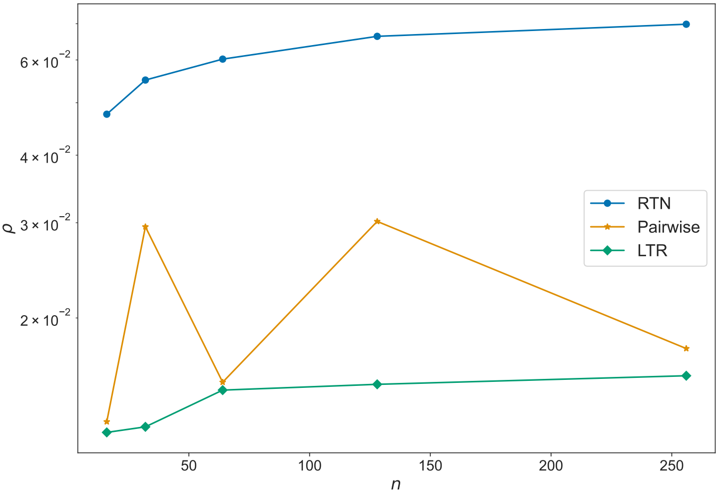


Comparison with the baseline in terms of t ($n = 256$ and $b = 3$)

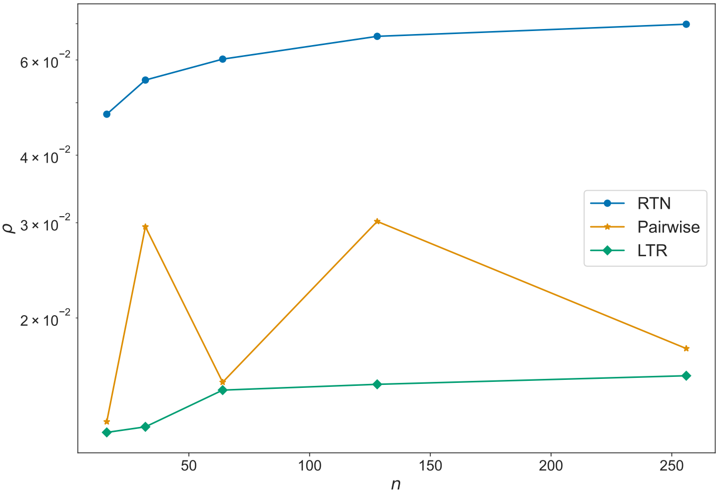


$\rightarrow$ For a given precision, the number of bits is reduced by 30% compared to RTN

On the FFT in terms of the dimension ($t = 4$ and $b = 3$)



On the FFT in terms of the dimension ($t = 4$ and $b = 3$)



$\rightarrow$ LTR is more accurate (≈ 10 times) and equally adapted for even/odd L

Conclusion

Wrap-up:

- Optimal complex-valued rank-one quantization algorithm
- Compared to RTN, the number of bits is reduced by 30% for a given precision on butterfly matrices

What's next?

- Short version available [Chaumette et al., 2025] and working paper soon to be released
- Quantization of a product of matrices of any rank
- Extend this work to quantize ReLU networks

Thanks for your attention

References

[1] V. Samborska. Scaling up: how increasing inputs has made artificial intelligence more capable. *Our World in Data*, 2025. https://ourworldindata.org/scaling-up-ai.

[2] B. Neyshabur, R. Tomioka, and N. Srebro. Norm-based capacity control in neural networks. In *Conference on learning theory*, pages 1376–1401. PMLR, 2015.

[3] R. Gribonval, T. Mary, and E. Riccietti. Optimal quantization of rank-one matrices in floating-point arithmetic—with applications to butterfly factorizations. 2023.

[4] J. W. Cooley and J. W. Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, 19 (90):297–301, 1965.

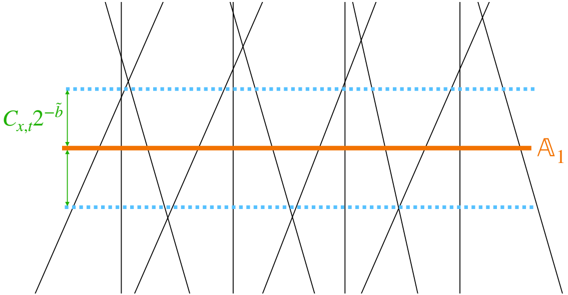
[5] M. Chaumette, R. Gribonval, and E. Riccietti. CROQuant: Complex Rank-One Quantization Algorithm. In *GRETSI 2025*, Strasbourg, France, August 2025.

Expression of f

$$f : \lambda \in \mathbb{C} \mapsto \max_{\hat{x} \in \mathrm{round}(\lambda x)} \|xy^H - \hat{x}\,\mathrm{round}(\mu(\hat{x})y)^H\|$$

where $\mu(\hat{x}) := \frac{\langle \hat{x}, x \rangle}{\|x\|^2}$ if $x \neq 0$ and 0 otherwise.

Towards an optimal stopping criterion



Typical behaviour near the accumulation lines.

Bound for the minimum value

Under mild assumptions on x, y and for any $b \geq \tilde{b}$, we have

$$\min\left(\min_{\mathbb{A}} f - L_{x,y,t}2^{-b}, f(\lambda_b)\right) \leq \min_{\mathbb{C}} f \leq f(\lambda_b)$$