FAST LEARNING OF FAST TRANSFORMS, WITH GUARANTEES

QUOC TUNG LE, LÉON ZHENG, ELISA RICCIETTI, RÉMI GRIBONVAL

Université de Lyon, ENS de Lyon, CNRS, INRIA, Laboratoire de l'Informatique du Parallélisme

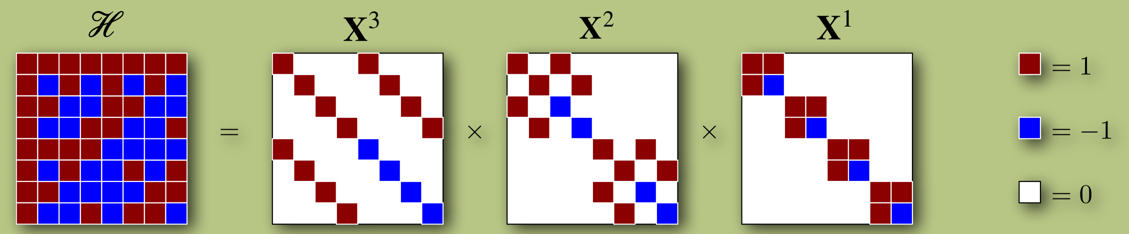
Valeo.ai, Paris

MULTI-LAYER BUTTERFLY MATRIX FACTORIZATION

PROBLEM: Approximate $\mathbf{A} \in \mathbb{C}^{2^J \times 2^J}$ by a product of J butterfly factors.

BUTTERFLY STRUCTURE: J factors $\mathbf{X}^J, \ldots, \mathbf{X}^1$ are called butterfly factors (**BF**) if we have $\mathbf{X}^\ell \in \mathbb{C}^{2^J \times 2^J}$ and $\text{supp}(\mathbf{X}^\ell) \subseteq \text{supp}(\mathbf{S}^\ell), \forall 1 \leq \ell \leq J$ where $\text{supp}(\mathbf{Z}) = \{(i,j) \mid \mathbf{Z}_{i,j} \neq 0\}$ and $\mathbf{S}^\ell := \mathbf{I}_{N/2^\ell} \otimes \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \otimes \mathbf{I}_{2^{\ell-1}} \in \{0,1\}^{2^J \times 2^J}$.

Example:



The Hadamard transform $\mathscr{H}$ and its butterfly factors

MATHEMATICAL FORMULATION:

$$\underset{\mathbf{X}^J,\ldots,\mathbf{X}^1}{\text{Minimize}} \; \|\mathbf{A} - \prod_{\ell=1}^{J} \mathbf{X}^\ell\|^2 \quad \text{such that } \mathbf{X}^J, \ldots, \mathbf{X}^1 \text{ are BF} \qquad (1)$$

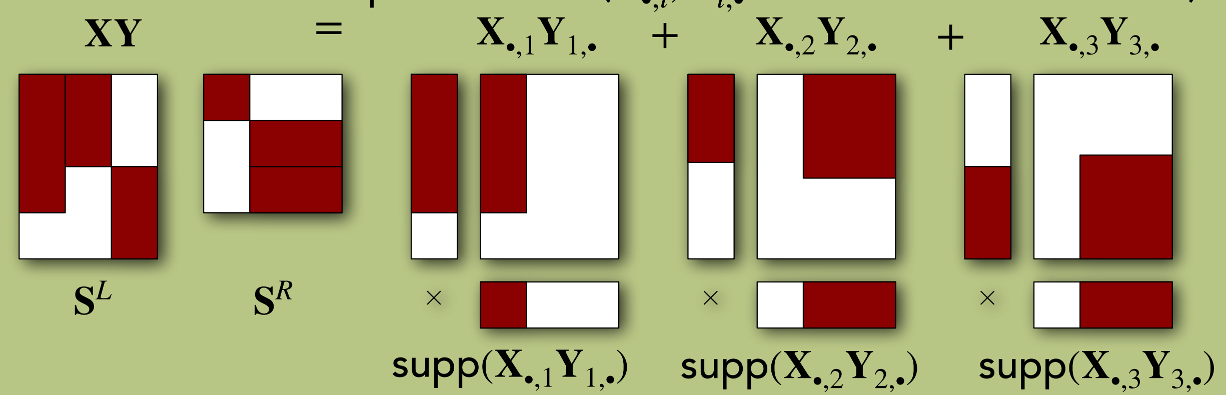
OBJECTIVE: An algorithm which is more efficient than classical gradient descent, with theoretical guarantee.

BACKGROUND: FIXED SUPPORT MATRIX FACTORIZATION

PROBLEM: Given $\mathbf{A} \in \mathbb{R}^{m \times n}, \mathbf{S}^L \in \{0,1\}^{m \times r}$ and $\mathbf{S}^R \in \{0,1\}^{r \times n}$:

$$\underset{(\mathbf{X},\mathbf{Y})}{\text{Minimize}} \quad \|\mathbf{A} - \mathbf{XY}\|_F^2, \text{ s.t } \text{supp}(\mathbf{X}) \subseteq \text{supp}(\mathbf{S}^L), \text{supp}(\mathbf{Y}) \subseteq \text{supp}(\mathbf{S}^R) \quad (2)$$

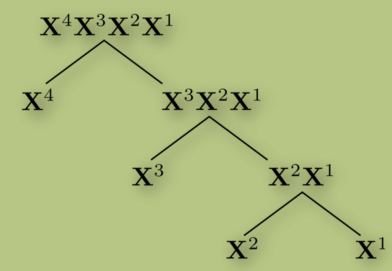
OBSERVATION: Decomposition $\mathbf{XY}$ ($\mathbf{Z}_{\bullet,i}, \mathbf{Z}_{i,\bullet}$ are ith column/row of $\mathbf{Z}$)



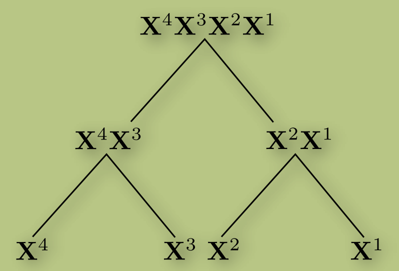
RESULT: If $\{\text{supp}(\mathbf{X}_{\bullet,i}\mathbf{Y}_{i,\bullet}) \mid i = 1, \ldots, r\}$ are pairwise disjoint, (2) is polynomially solvable and identifiable [1,2].

MAIN CONTRIBUTION: HIERARCHICAL METHOD

MAIN IDEA: Recursively use algorithms for (2) to factorize a matrix into two factors at each intermediate level of the hierarchy.



Unbalanced



Balanced

Two strategies to perform hierarchical factorization. Each corresponds to a tree

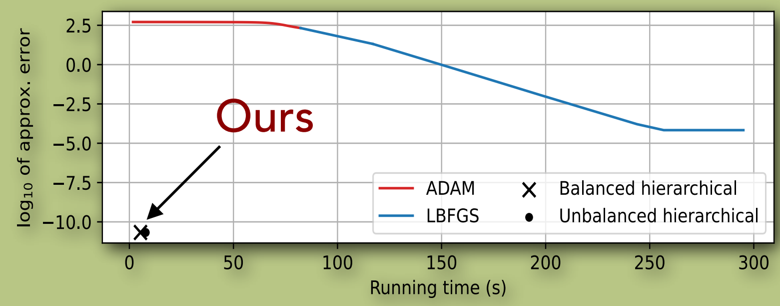
INPUT: A matrix $\mathbf{Z} = \mathbf{X}^J \ldots \mathbf{X}^1$ where $\mathbf{X}^J, \ldots, \mathbf{X}^1$ are butterfly factors.

THEORETICAL GUARANTEE: For any choice of tree, the hierarchical method will yield J factors $\bar{\mathbf{X}}^J, \ldots, \bar{\mathbf{X}}^1$ such that:

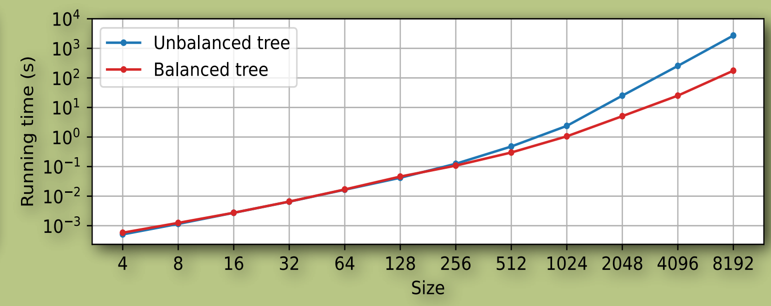
1) Exact factorization: $\mathbf{Z} = \bar{\mathbf{X}}^J \ldots \bar{\mathbf{X}}^1$

2) Recovery: $\mathbf{X}^\ell = \mathbf{D}^{\ell-1}\bar{\mathbf{X}}^\ell(\mathbf{D}^\ell)^{-1}$ where $\mathbf{D}^J, \ldots, \mathbf{D}^0$ are invertible diagonal matrices satisfying $\mathbf{D}^0 = \mathbf{D}^J = \mathbf{I}_{2^J}$.

EXPERIMENTAL RESULT: Comparison between the state-of-the-art method [3] (ADAM + LBFGS) and our methods (unbalanced and balanced).



Precision and running time of [3] and our methods in the factorization of the Discrete Fourier Transform of size 512 ($J = 9$)



Running time of *balanced* and *unbalanced* strategies factorizing a noisy version of a product of J butterfly factors, $1 \leq J \leq 13$.

REFERENCES

[1]: Q.T. Le, E. Riccietti and R. Gribonval, "Spurious Valleys, Spurious Minima and NP-hardness of Sparse Matrix Factorization With Fixed Support", arXiv preprint arXiv:2112.00386, 2021.

[2]: L.Zheng, E. Riccietti and R. Gribonval, "Efficient Identification of Butterfly Sparse Matrix Factorization", arXiv preprint, arXiv:2110.01235, 2022.

[3]: T. Dao, A. Gu, M. Eichhorn, A. Rudra and C. Re, "Learning Fast Algorithms for Linear Transforms Using Butterfly Factorization", 36th International Conference of Machine Learning, June 2019.