Efficient Identification of Butterfly Sparse Matrix Factorizations

Léon Zheng, Elisa Riccietti, Rémi Gribonval

March 2 - SIAM CSE 2023

Sparse matrix factorization

Given a matrix $\mathbf{Z}$ and $J \geq 2$, find sparse factors $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(J)}$ such that

$$\mathbf{Z} \approx \mathbf{X}^{(1)}\mathbf{X}^{(2)} \ldots \mathbf{X}^{(J)}.$$

Sparse matrix factorization

Given a matrix $\mathbf{Z}$ and $J \geq 2$, find sparse factors $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(J)}$ such that

$$\mathbf{Z} \approx \mathbf{X}^{(1)}\mathbf{X}^{(2)} \ldots \mathbf{X}^{(J)}.$$

Fast matrix-vector multiplication: $\mathbf{Z}\mathbf{x} \approx \mathbf{X}^{(1)}\mathbf{X}^{(2)} \ldots \mathbf{X}^{(J)}\mathbf{x}$.

Sparse matrix factorization

Given a matrix $\mathbf{Z}$ and $J \geq 2$, find sparse factors $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(J)}$ such that

$$\mathbf{Z} \approx \mathbf{X}^{(1)}\mathbf{X}^{(2)} \ldots \mathbf{X}^{(J)}.$$

Fast matrix-vector multiplication: $\mathbf{Z}\mathbf{x} \approx \mathbf{X}^{(1)}\mathbf{X}^{(2)} \ldots \mathbf{X}^{(J)}\mathbf{x}$.

Example (Fourier integral operators)

Fast evaluation of $u(x) = \sum_{k \in \Omega} e^{2\pi i \Phi(x,k)} f(k)$, for all $x \in X$?

Sparse matrix factorization

Given a matrix $\mathbf{Z}$ and $J \geq 2$, find sparse factors $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(J)}$ such that

$$\mathbf{Z} \approx \mathbf{X}^{(1)} \mathbf{X}^{(2)} \ldots \mathbf{X}^{(J)}.$$

Fast matrix-vector multiplication: $\mathbf{Z}\mathbf{x} \approx \mathbf{X}^{(1)} \mathbf{X}^{(2)} \ldots \mathbf{X}^{(J)} \mathbf{x}$.

Example (Fourier integral operators)

Fast evaluation of $u(x) = \sum_{k \in \Omega} e^{2\pi i \Phi(x,k)} f(k)$, for all $x \in X$?
Factorize $\mathbf{K} := (e^{2\pi i \Phi(x,k)})_{x \in X, k \in \Omega} \approx \mathbf{X}^{(1)} \ldots \mathbf{X}^{(J)}$.

[E. Candes et al., **A fast butterfly algorithm for the computation of Fourier integral operators**, MMS, 2009]

Sparse matrix factorization

Given a matrix $\mathbf{Z}$ and $J \geq 2$, find sparse factors $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(J)}$ such that

$$\mathbf{Z} \approx \mathbf{X}^{(1)}\mathbf{X}^{(2)} \ldots \mathbf{X}^{(J)}.$$

Fast matrix-vector multiplication: $\mathbf{Z}\mathbf{x} \approx \mathbf{X}^{(1)}\mathbf{X}^{(2)} \ldots \mathbf{X}^{(J)}\mathbf{x}$.

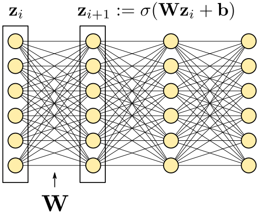
Example (Fourier integral operators)

Fast evaluation of $u(x) = \sum_{k\in\Omega} e^{2\pi i \Phi(x,k)} f(k)$, for all $x \in X$?
Factorize $\mathbf{K} := (e^{2\pi i \Phi(x,k)})_{x\in X, k\in\Omega} \approx \mathbf{X}^{(1)} \ldots \mathbf{X}^{(J)}$.

[E. Candes et al., **A fast butterfly algorithm for the computation of Fourier integral operators**, MMS, 2009]

Example (Compress neural network)

Factorize weight matrices in overparameterized neural networks.



[T. Dao et al., **Monarch: Expressive structured matrices for efficient and accurate training**, ICML, 2022]

Sparse matrix factorization

Given a matrix $\mathbf{Z}$ and $J \geq 2$, find sparse factors $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(J)}$ such that

$$\mathbf{Z} \approx \mathbf{X}^{(1)}\mathbf{X}^{(2)} \ldots \mathbf{X}^{(J)}.$$

Fast matrix-vector multiplication: $\mathbf{Zx} \approx \mathbf{X}^{(1)}\mathbf{X}^{(2)} \ldots \mathbf{X}^{(J)}\mathbf{x}$.

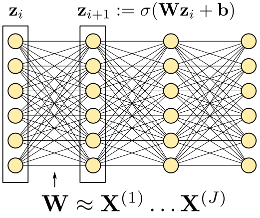
Example (Fourier integral operators)

Fast evaluation of $u(x) = \sum_{k \in \Omega} e^{2\pi i \Phi(x,k)} f(k)$, for all $x \in X$?
Factorize $\mathbf{K} := (e^{2\pi i \Phi(x,k)})_{x \in X, k \in \Omega} \approx \mathbf{X}^{(1)} \ldots \mathbf{X}^{(J)}$.

[E. Candes et al., **A fast butterfly algorithm for the computation of Fourier integral operators**, MMS, 2009]

Example (Compress neural network)

Factorize weight matrices in overparameterized neural networks.



[T. Dao et al., **Monarch: Expressive structured matrices for efficient and accurate training**, ICML, 2022]

How to factorize a matrix into sparse factors?

Problem formulation

$$\min_{\mathbf{X}^{(1)},\dots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\dots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \{\mathbf{X}^{(\ell)}\}_\ell \text{ are sparse.}$$

Choices for sparsity constraint?

How to factorize a matrix into sparse factors?

Problem formulation

$$\min_{\mathbf{X}^{(1)},\ldots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\ldots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \{\mathbf{X}^{(\ell)}\}_\ell \text{ are sparse.}$$

Choices for sparsity constraint?

1. **Classical sparsity patterns**: k-sparsity by column and/or by row

How to factorize a matrix into sparse factors?

Problem formulation

$$\min_{\mathbf{X}^{(1)},\ldots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\ldots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \{\mathbf{X}^{(\ell)}\}_\ell \text{ are sparse.}$$

Choices for sparsity constraint?

1. **Classical sparsity patterns**: k-sparsity by column and/or by row
2. **Fixed-support constraint**: $\text{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}^{(\ell)}$ for $\ell = 1, \ldots, J$.

How to factorize a matrix into sparse factors?

Problem formulation

$$\min_{\mathbf{X}^{(1)},\ldots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\ldots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \{\mathbf{X}^{(\ell)}\}_\ell \text{ are sparse.}$$

Choices for sparsity constraint?

1. **Classical sparsity patterns**: k-sparsity by column and/or by row
2. **Fixed-support constraint**: $\text{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}^{(\ell)}$ for $\ell = 1, \ldots, J$.

A difficult problem

How to factorize a matrix into sparse factors?

Problem formulation

$$\min_{\mathbf{X}^{(1)},\ldots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\ldots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \{\mathbf{X}^{(\ell)}\}_\ell \text{ are sparse.}$$

Choices for sparsity constraint?

1. **Classical sparsity patterns**: k-sparsity by column and/or by row
2. **Fixed-support constraint**: $\text{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}^{(\ell)}$ for $\ell = 1, \ldots, J$.

A difficult problem

- Sparse coding is NP-hard.
  [S. Foucart, H. Rauhut, **A mathematical introduction to compressive sensing**, ANHA, 2013]

How to factorize a matrix into sparse factors?

Problem formulation

$$\min_{\mathbf{X}^{(1)},\ldots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\ldots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \{\mathbf{X}^{(\ell)}\}_\ell \text{ are sparse.}$$

Choices for sparsity constraint?

1. **Classical sparsity patterns**: k-sparsity by column and/or by row
2. **Fixed-support constraint**: $\text{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}^{(\ell)}$ for $\ell = 1, \ldots, J$.

A difficult problem

- Sparse coding is NP-hard.
  [S. Foucart, H. Rauhut, **A mathematical introduction to compressive sensing**, ANHA, 2013]
- Fixed-support setting is NP-hard for $J = 2$ factors.
  [Le et al., **Spurious valleys, NP-hardness, and tractability of sparse matrix fact. with fixed support**, SIMAX, 2022]

How to factorize a matrix into sparse factors?

Problem formulation

$$\min_{\mathbf{X}^{(1)},\dots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\dots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \{\mathbf{X}^{(\ell)}\}_\ell \text{ are sparse.}$$

Choices for sparsity constraint?

1. **Classical sparsity patterns**: k-sparsity by column and/or by row
2. **Fixed-support constraint**: $\operatorname{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}^{(\ell)}$ for $\ell = 1, \dots, J$.

A difficult problem

- Sparse coding is NP-hard.
  [S. Foucart, H. Rauhut, **A mathematical introduction to compressive sensing**, ANHA, 2013]
- Fixed-support setting is NP-hard for $J = 2$ factors.
  [Le et al., **Spurious valleys, NP-hardness, and tractability of sparse matrix fact. with fixed support**, SIMAX, 2022]
- Gradient-based methods lack guarantees.

How to factorize a matrix into sparse factors?

Problem formulation

$$\min_{\mathbf{X}^{(1)},\ldots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\ldots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \{\mathbf{X}^{(\ell)}\}_\ell \text{ are sparse.}$$

Choices for sparsity constraint?

1. **Classical sparsity patterns**: k-sparsity by column and/or by row
2. **Fixed-support constraint**: $\mathrm{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}^{(\ell)}$ for $\ell = 1, \ldots, J$.

A difficult problem

- Sparse coding is NP-hard.
  [S. Foucart, H. Rauhut, **A mathematical introduction to compressive sensing**, ANHA, 2013]
- Fixed-support setting is NP-hard for $J = 2$ factors.
  [Le et al., **Spurious valleys, NP-hardness, and tractability of sparse matrix fact. with fixed support**, SIMAX, 2022]
- Gradient-based methods lack guarantees.

When is the problem well-posed? Uniqueness of solution? Stability?
$\rightarrow$ Still an open question.

What makes the butterfly structure a good choice?

Case of square matrices of size $N = 2^J$.

What makes the butterfly structure a good choice?

Case of square matrices of size $N = 2^J$.

Definition (Butterfly structure)

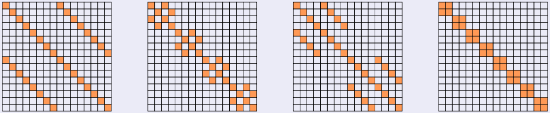
For each $\ell \in [J]$, $\mathrm{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(\ell)}$

What makes the butterfly structure a good choice?

Case of square matrices of size $N = 2^J$.

Definition (Butterfly structure)

For each $\ell \in [J]$, $\mathrm{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(\ell)}$, where $\mathbf{S}_{\mathtt{bf}}^{(\ell)} := \mathbf{I}_{2^{\ell-1}} \otimes \left[\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix}\right] \otimes \mathbf{I}_{N/2^{\ell}}$.



$\mathbf{S}_{\mathtt{bf}}^{(1)}$ $\mathbf{S}_{\mathtt{bf}}^{(2)}$ $\mathbf{S}_{\mathtt{bf}}^{(3)}$ $\mathbf{S}_{\mathtt{bf}}^{(4)}$

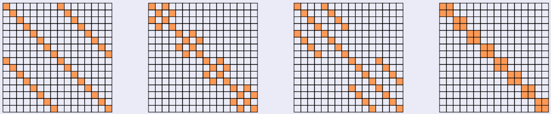
Example for $N = 16$, with 4 factors.

What makes the butterfly structure a good choice?

Case of square matrices of size $N = 2^J$.

Definition (Butterfly structure)

For each $\ell \in [J]$, $\mathrm{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(\ell)}$, where $\mathbf{S}_{\mathtt{bf}}^{(\ell)} := \mathbf{I}_{\mathbf{2}^{\ell-1}} \otimes \left[\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix}\right] \otimes \mathbf{I}_{\mathbf{N}/\mathbf{2}^{\ell}}$.



$\mathbf{S}_{\mathtt{bf}}^{(1)}$ $\mathbf{S}_{\mathtt{bf}}^{(2)}$ $\mathbf{S}_{\mathtt{bf}}^{(3)}$ $\mathbf{S}_{\mathtt{bf}}^{(4)}$

Example for $N = 16$, with 4 factors.

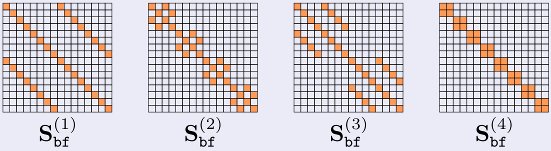
1. Enables fast $\mathcal{O}(N \log N)$ matrix-vector multiplication

What makes the butterfly structure a good choice?

Case of square matrices of size $N = 2^J$.

Definition (Butterfly structure)

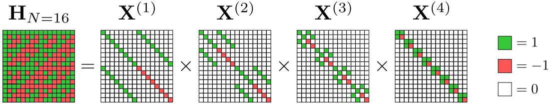
For each $\ell \in [J]$, $\text{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}_{\text{bf}}^{(\ell)}$, where $\mathbf{S}_{\text{bf}}^{(\ell)} := \mathbf{I}_{2^{\ell-1}} \otimes \left[\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix}\right] \otimes \mathbf{I}_{N/2^\ell}$.



Example for $N = 16$, with 4 factors.

1. Enables fast $\mathcal{O}(N \log N)$ matrix-vector multiplication
2. Captures common fast transforms (Hadamard, DFT, DCT, ...)

[T. Dao et al., **Kaleidoscope: An efficient, learnable representation for all structured linear maps**, ICLR, 2020]

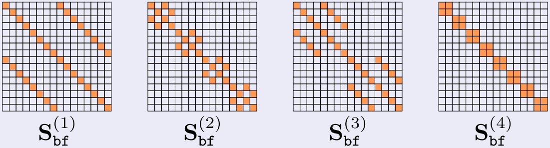


What makes the butterfly structure a good choice?

Case of square matrices of size $N = 2^J$.

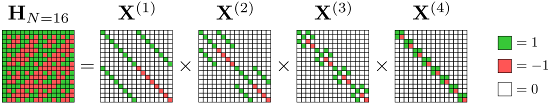
Definition (Butterfly structure)

For each $\ell \in [J]$, $\text{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}_{\texttt{bf}}^{(\ell)}$, where $\mathbf{S}_{\texttt{bf}}^{(\ell)} := \mathbf{I}_{2^{\ell-1}} \otimes \left[\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix}\right] \otimes \mathbf{I}_{N/2^{\ell}}$.



Example for $N = 16$, with 4 factors.

1. Enables fast $\mathcal{O}(N \log N)$ matrix-vector multiplication
2. Captures common fast transforms (Hadamard, DFT, DCT, ...)
   [T. Dao et al., **Kaleidoscope: An efficient, learnable representation for all structured linear maps**, ICLR, 2020]



3. Makes the sparse matrix factorization problem well-posed
   [L. Zheng et al., **Efficient identification of butterfly sparse matrix factorizations**, SIMODS, 2023]

Overview of this talk

Butterfly factorization problem

$$\min_{\mathbf{X}^{(1)},\ldots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\ldots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \mathrm{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(\ell)}.$$

Overview of this talk

Butterfly factorization problem

$$\min_{\mathbf{X}^{(1)},\ldots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\ldots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \operatorname{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(\ell)}.$$

- Gradient-descent method
  [Le Magouarou et al., **Flexible multilayer sparse approximations of matrices and applications**, JSTSP, 2016]
  [T. Dao et al., **Learning fast algorithms for linear transforms using butterfly factorizations**, ICML, 2019.]
- Hierarchical factorization (butterfly algorithms)
  [Y. Li et al., **Butterfly factorization**, MMS, 2015]

Overview of this talk

Butterfly factorization problem

$$\min_{\mathbf{X}^{(1)},\ldots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\ldots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \mathrm{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(\ell)}.$$

- Gradient-descent method
  [Le Magouarou et al., **Flexible multilayer sparse approximations of matrices and applications**, JSTSP, 2016]
  [T. Dao et al., **Learning fast algorithms for linear transforms using butterfly factorizations**, ICML, 2019.]
- Hierarchical factorization (butterfly algorithms)
  [Y. Li et al., **Butterfly factorization**, MMS, 2015]

Contributions

Overview of this talk

Butterfly factorization problem

$$\min_{\mathbf{X}^{(1)},\dots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\dots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \operatorname{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(\ell)}.$$

- Gradient-descent method
  [Le Magouarou et al., **Flexible multilayer sparse approximations of matrices and applications**, JSTSP, 2016]
  [T. Dao et al., **Learning fast algorithms for linear transforms using butterfly factorizations**, ICML, 2019.]
- Hierarchical factorization (butterfly algorithms)
  [Y. Li et al., **Butterfly factorization**, MMS, 2015]

Contributions

1. We prove that the butterfly factorization is essentially unique.

$$\mathbf{X}^{(1)}\dots\mathbf{X}^{(J)} = \bar{\mathbf{X}}^{(1)}\dots\bar{\mathbf{X}}^{(J)} \implies (\mathbf{X}^{(\ell)})_{\ell=1}^{J} \sim (\bar{\mathbf{X}}^{(\ell)})_{\ell=1}^{J}$$

Overview of this talk

Butterfly factorization problem

$$\min_{\mathbf{X}^{(1)},\ldots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\ldots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \operatorname{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(\ell)}.$$

- Gradient-descent method
  [Le Magouarou et al., **Flexible multilayer sparse approximations of matrices and applications**, JSTSP, 2016]
  [T. Dao et al., **Learning fast algorithms for linear transforms using butterfly factorizations**, ICML, 2019.]
- Hierarchical factorization (butterfly algorithms)
  [Y. Li et al., **Butterfly factorization**, MMS, 2015]

Contributions

1. We prove that the butterfly factorization is essentially unique.

$$\mathbf{X}^{(1)}\ldots\mathbf{X}^{(J)} = \bar{\mathbf{X}}^{(1)}\ldots\bar{\mathbf{X}}^{(J)} \implies (\mathbf{X}^{(\ell)})_{\ell=1}^{J} \sim (\bar{\mathbf{X}}^{(\ell)})_{\ell=1}^{J}$$

2. Factors are recovered by a flexible hierarchical factorization algorithm.

Overview of this talk

Butterfly factorization problem

$$\min_{\mathbf{X}^{(1)},\ldots,\mathbf{X}^{(J)}} \left\| \mathbf{Z} - \mathbf{X}^{(1)}\mathbf{X}^{(2)}\ldots\mathbf{X}^{(J)} \right\|_F, \quad \text{such that } \operatorname{supp}(\mathbf{X}^{(\ell)}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(\ell)}.$$

- Gradient-descent method
  [Le Magouarou et al., **Flexible multilayer sparse approximations of matrices and applications**, JSTSP, 2016]
  [T. Dao et al., **Learning fast algorithms for linear transforms using butterfly factorizations**, ICML, 2019.]
- Hierarchical factorization (butterfly algorithms)
  [Y. Li et al., **Butterfly factorization**, MMS, 2015]

Contributions

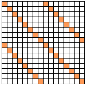
1. We prove that the butterfly factorization is essentially unique.

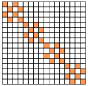
$$\mathbf{X}^{(1)}\ldots\mathbf{X}^{(J)} = \bar{\mathbf{X}}^{(1)}\ldots\bar{\mathbf{X}}^{(J)} \implies (\mathbf{X}^{(\ell)})_{\ell=1}^{J} \sim (\bar{\mathbf{X}}^{(\ell)})_{\ell=1}^{J}$$

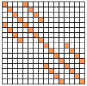
2. Factors are recovered by a flexible hierarchical factorization algorithm.
3. The algo. is numerically faster, more accurate than gradient-descent.

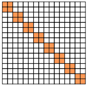
Hierarchical factorization algorithm

Let $\mathbf{Z} := \mathbf{X}^{(1)}\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}$ such that:

$\mathrm{supp}(\mathbf{X}^{(1)}) \subseteq$ 

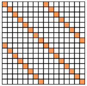
$\mathrm{supp}(\mathbf{X}^{(3)}) \subseteq$ 

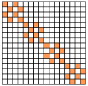
$\mathrm{supp}(\mathbf{X}^{(2)}) \subseteq$ 

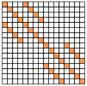
$\mathrm{supp}(\mathbf{X}^{(4)}) \subseteq$ 

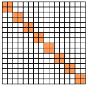
Hierarchical factorization algorithm

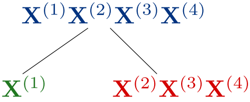
Let $\mathbf{Z} := \mathbf{X}^{(1)}\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}$ such that:

$\mathrm{supp}(\mathbf{X}^{(1)}) \subseteq$ 

$\mathrm{supp}(\mathbf{X}^{(3)}) \subseteq$ 

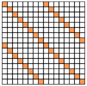
$\mathrm{supp}(\mathbf{X}^{(2)}) \subseteq$ 

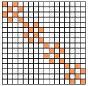
$\mathrm{supp}(\mathbf{X}^{(4)}) \subseteq$ 

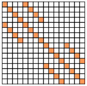


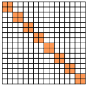
Hierarchical factorization algorithm

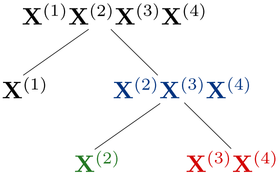
Let $\mathbf{Z} := \mathbf{X}^{(1)}\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}$ such that:

$\mathrm{supp}(\mathbf{X}^{(1)}) \subseteq$ 

$\mathrm{supp}(\mathbf{X}^{(3)}) \subseteq$ 

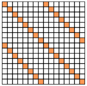
$\mathrm{supp}(\mathbf{X}^{(2)}) \subseteq$ 

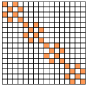
$\mathrm{supp}(\mathbf{X}^{(4)}) \subseteq$ 

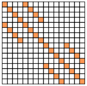


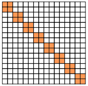
Hierarchical factorization algorithm

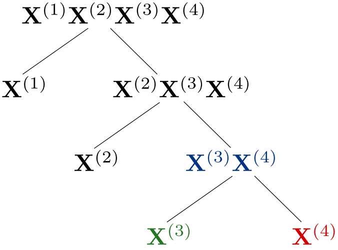
Let $\mathbf{Z} := \mathbf{X}^{(1)}\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}$ such that:

$\mathrm{supp}(\mathbf{X}^{(1)}) \subseteq$ 

$\mathrm{supp}(\mathbf{X}^{(3)}) \subseteq$ 

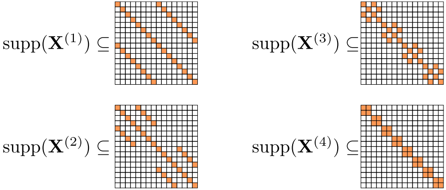
$\mathrm{supp}(\mathbf{X}^{(2)}) \subseteq$ 

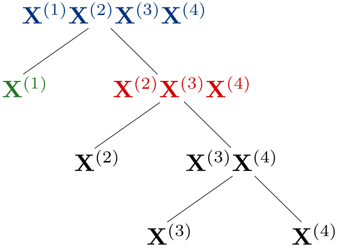
$\mathrm{supp}(\mathbf{X}^{(4)}) \subseteq$ 



Hierarchical factorization algorithm

Let $\mathbf{Z} := \mathbf{X}^{(1)}\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}$ such that:

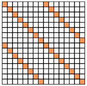


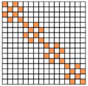


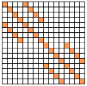
How to recover the partial products?

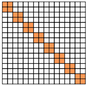
Hierarchical factorization algorithm

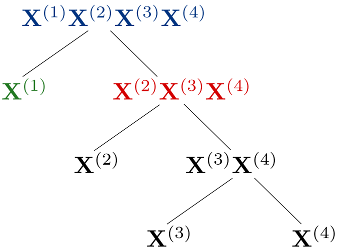
Let $\mathbf{Z} := \mathbf{X}^{(1)}\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}$ such that:

$\mathrm{supp}(\mathbf{X}^{(1)}) \subseteq$ 

$\mathrm{supp}(\mathbf{X}^{(3)}) \subseteq$ 

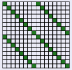
$\mathrm{supp}(\mathbf{X}^{(2)}) \subseteq$ 

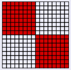
$\mathrm{supp}(\mathbf{X}^{(4)}) \subseteq$ 



How to recover the partial products? $\rightarrow$ use their known supports

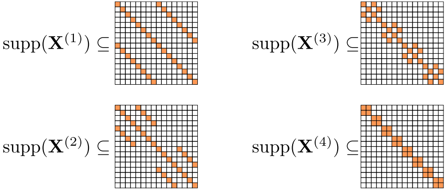
Lemma (Supports of the partial products)

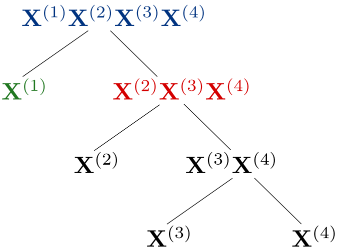
$\mathrm{supp}(\mathbf{X}^{(1)}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(1)}$

$\mathrm{supp}(\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)}$

Hierarchical factorization algorithm

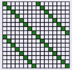
Let $\mathbf{Z} := \mathbf{X}^{(1)}\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}$ such that:

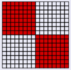




How to recover the partial products? $\rightarrow$ use their known supports

Lemma (Supports of the partial products)

$\mathrm{supp}(\mathbf{X}^{(1)}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(1)}$

$\mathrm{supp}(\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)}$

Two-layer fixed-support problem:

$$\min_{\mathbf{A},\mathbf{B}} \|\mathbf{Z} - \mathbf{A}\mathbf{B}\|_F \,, \text{ s.t. } \mathrm{supp}(\mathbf{A}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(1)}, \ \mathrm{supp}(\mathbf{B}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)} \qquad (\star)$$

Two-layer fixed-support sparse matrix factorization

$$\min_{\mathbf{A},\mathbf{B}} \|\mathbf{Z} - \mathbf{A}\mathbf{B}\|_F \,, \text{ s.t.} \operatorname{supp}(\mathbf{A}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(1)}, \operatorname{supp}(\mathbf{B}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)} \qquad (\star)$$

Two-layer fixed-support sparse matrix factorization

$$\min_{\mathbf{A},\mathbf{B}} \|\mathbf{Z} - \mathbf{A}\mathbf{B}\|_F\,, \text{ s.t. } \operatorname{supp}(\mathbf{A}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(1)},\ \operatorname{supp}(\mathbf{B}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)} \qquad (\star)$$

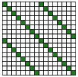
Fact: $\mathbf{AB} = \sum_{i=1}^{N} \mathbf{A}_{\bullet,i}\mathbf{B}_{i,\bullet}$.

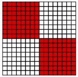
Two-layer fixed-support sparse matrix factorization

$$\min_{\mathbf{A},\mathbf{B}} \|\mathbf{Z} - \mathbf{A}\mathbf{B}\|_F \,, \text{ s.t.} \operatorname{supp}(\mathbf{A}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(1)}, \ \operatorname{supp}(\mathbf{B}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)} \qquad (\star)$$

Fact: $\mathbf{AB} = \sum_{i=1}^{N} \mathbf{A}_{\bullet,i}\mathbf{B}_{i,\bullet}$.

Constraint on the pair of factors

$\operatorname{supp}(\mathbf{A}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(1)}$

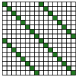
$\operatorname{supp}(\mathbf{B}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)}$

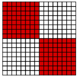
Two-layer fixed-support sparse matrix factorization

$$\min_{\mathbf{A},\mathbf{B}} \|\mathbf{Z} - \mathbf{A}\mathbf{B}\|_F \,, \text{ s.t. } \operatorname{supp}(\mathbf{A}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(1)}, \ \operatorname{supp}(\mathbf{B}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)} \qquad (\star)$$

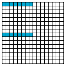
Fact: $\mathbf{AB} = \sum_{i=1}^{N} \mathbf{A}_{\bullet,i}\mathbf{B}_{i,\bullet}$.

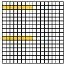
Constraint on the pair of factors

$\operatorname{supp}(\mathbf{A}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(1)}$

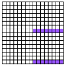
$\operatorname{supp}(\mathbf{B}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)}$

Constraint on the rank-one matrices

$\operatorname{supp}(\mathbf{A}_{\bullet,1}\mathbf{B}_{1,\bullet}) \subseteq$  $= \mathcal{S}_1$

$\operatorname{supp}(\mathbf{A}_{\bullet,2}\mathbf{B}_{2,\bullet}) \subseteq$  $= \mathcal{S}_2$

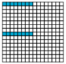
$\vdots$

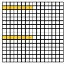
$\operatorname{supp}(\mathbf{A}_{\bullet,N}\mathbf{B}_{N,\bullet}) \subseteq$  $= \mathcal{S}_N$

Two-layer fixed-support sparse matrix factorization

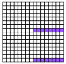
$$\min_{\mathbf{A},\mathbf{B}} \|\mathbf{Z} - \mathbf{A}\mathbf{B}\|_F \,, \text{ s.t. } \mathrm{supp}(\mathbf{A}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(1)}, \; \mathrm{supp}(\mathbf{B}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)} \qquad (\star)$$

Constraint on the rank-one matrices

$\mathrm{supp}(\mathbf{A}_{\bullet,1}\mathbf{B}_{1,\bullet}) \subseteq$  $= \mathcal{S}_1$

$\mathrm{supp}(\mathbf{A}_{\bullet,2}\mathbf{B}_{2,\bullet}) \subseteq$  $= \mathcal{S}_2$

$\cdots$

$\mathrm{supp}(\mathbf{A}_{\bullet,N}\mathbf{B}_{N,\bullet}) \subseteq$  $= \mathcal{S}_N$

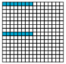
Theorem ([Le et al. 2022; Zheng et al. 2023])

The rank-one matrices have **pairwise disjoint supports**.

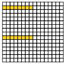
Two-layer fixed-support sparse matrix factorization

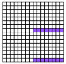
$$\min_{\mathbf{A},\mathbf{B}} \|\mathbf{Z} - \mathbf{A}\mathbf{B}\|_F, \text{ s.t. } \operatorname{supp}(\mathbf{A}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(1)}, \ \operatorname{supp}(\mathbf{B}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)} \qquad (\star)$$

Constraint on the rank-one matrices

$\operatorname{supp}(\mathbf{A}_{\bullet,1}\mathbf{B}_{1,\bullet}) \subseteq$  $= \mathcal{S}_1$

$\cdots$

$\operatorname{supp}(\mathbf{A}_{\bullet,2}\mathbf{B}_{2,\bullet}) \subseteq$  $= \mathcal{S}_2$

$\operatorname{supp}(\mathbf{A}_{\bullet,N}\mathbf{B}_{N,\bullet}) \subseteq$  $= \mathcal{S}_N$

Theorem ([Le et al. 2022; Zheng et al. 2023])

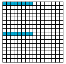
The rank-one matrices have **pairwise disjoint supports**. *Consequently:*

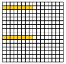
1. *there exists a polynomial algorithm to find an optimal solution to* $(\star)$

Two-layer fixed-support sparse matrix factorization

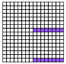
$$\min_{\mathbf{A},\mathbf{B}} \|\mathbf{Z} - \mathbf{A}\mathbf{B}\|_F \,, \text{ s.t. } \mathrm{supp}(\mathbf{A}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(1)}, \; \mathrm{supp}(\mathbf{B}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)} \qquad (\star)$$

Constraint on the rank-one matrices

$\mathrm{supp}(\mathbf{A}_{\bullet,1}\mathbf{B}_{1,\bullet}) \subseteq$  $= \mathcal{S}_1$

$\mathrm{supp}(\mathbf{A}_{\bullet,2}\mathbf{B}_{2,\bullet}) \subseteq$  $= \mathcal{S}_2$

$\cdots$

$\mathrm{supp}(\mathbf{A}_{\bullet,N}\mathbf{B}_{N,\bullet}) \subseteq$  $= \mathcal{S}_N$

Theorem ([Le et al. 2022; Zheng et al. 2023])

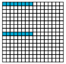
The rank-one matrices have **pairwise disjoint supports***. Consequently:*

1. *there exists a polynomial algorithm to find an optimal solution to* $(\star)$
2. *the solution is essentially unique in the noiseless setting.*

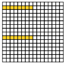
Two-layer fixed-support sparse matrix factorization

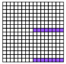
$$\min_{\mathbf{A},\mathbf{B}} \|\mathbf{Z} - \mathbf{AB}\|_F \,, \text{ s.t. } \mathrm{supp}(\mathbf{A}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(1)}, \ \mathrm{supp}(\mathbf{B}) \subseteq \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)} \qquad (\star)$$

Constraint on the rank-one matrices

$\mathrm{supp}(\mathbf{A}_{\bullet,1}\mathbf{B}_{1,\bullet}) \subseteq$  $= \mathcal{S}_1$

$\cdots$

$\mathrm{supp}(\mathbf{A}_{\bullet,2}\mathbf{B}_{2,\bullet}) \subseteq$  $= \mathcal{S}_2$

$\mathrm{supp}(\mathbf{A}_{\bullet,N}\mathbf{B}_{N,\bullet}) \subseteq$  $= \mathcal{S}_N$

Theorem ([Le et al. 2022; Zheng et al. 2023])

The rank-one matrices have **pairwise disjoint supports***. Consequently:*

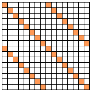
1. *there exists a polynomial algorithm to find an optimal solution to* $(\star)$
2. *the solution is essentially unique in the noiseless setting.*

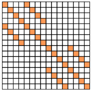
Algorithm:

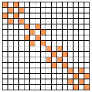
1. Extract the submatrices $\mathbf{Z}_{|\mathcal{S}_i}$, $i = 1, \ldots, N$
2. Perform best rank-one approximation for each submatrix

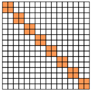
Hierarchical factorization algorithm

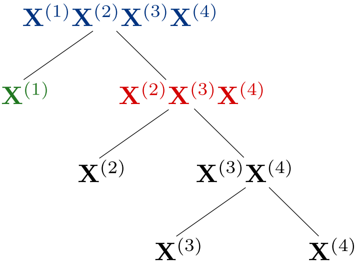
Let $\mathbf{Z} := \mathbf{X}^{(1)}\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}$ such that:

$\mathrm{supp}(\mathbf{X}^{(1)}) \subseteq$ 

$\mathrm{supp}(\mathbf{X}^{(2)}) \subseteq$ 

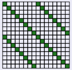
$\mathrm{supp}(\mathbf{X}^{(3)}) \subseteq$ 

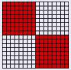
$\mathrm{supp}(\mathbf{X}^{(4)}) \subseteq$ 



The two-layer procedure is repeated **recursively**.

Lemma (Support of the partial products)

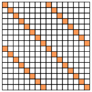
$\mathrm{supp}(\mathbf{X}^{(1)}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(1)}$

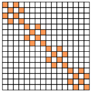
$\mathrm{supp}(\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(2)}\mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)}$

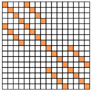
Corresponding rank-one supports are pairwise disjoint.

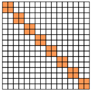
Hierarchical factorization algorithm

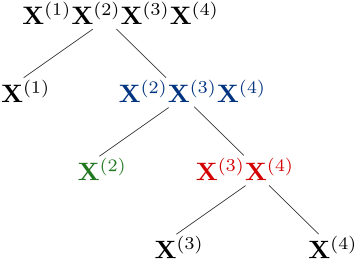
Let $\mathbf{Z} := \mathbf{X}^{(1)}\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}$ such that:

$\mathrm{supp}(\mathbf{X}^{(1)}) \subseteq$ 

$\mathrm{supp}(\mathbf{X}^{(3)}) \subseteq$ 

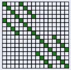
$\mathrm{supp}(\mathbf{X}^{(2)}) \subseteq$ 

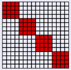
$\mathrm{supp}(\mathbf{X}^{(4)}) \subseteq$ 



The two-layer procedure is repeated **recursively**.

Lemma (Support of the partial products)

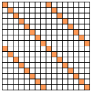
$\mathrm{supp}(\mathbf{X}^{(2)}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(2)}$

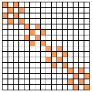
$\mathrm{supp}(\mathbf{X}^{(3)}\mathbf{X}^{(4)}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(3)}\mathbf{S}_{\mathtt{bf}}^{(4)}$

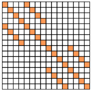
Corresponding rank-one supports are pairwise disjoint.

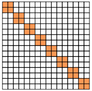
Hierarchical factorization algorithm

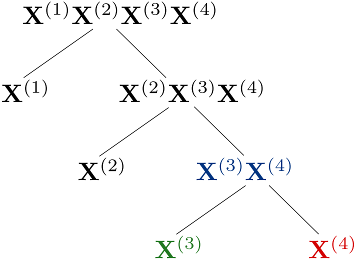
Let $\mathbf{Z} := \mathbf{X}^{(1)}\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}$ such that:

$\mathrm{supp}(\mathbf{X}^{(1)}) \subseteq$ 

$\mathrm{supp}(\mathbf{X}^{(3)}) \subseteq$ 

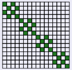
$\mathrm{supp}(\mathbf{X}^{(2)}) \subseteq$ 

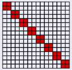
$\mathrm{supp}(\mathbf{X}^{(4)}) \subseteq$ 



The two-layer procedure is repeated **recursively**.

Lemma (Support of the partial products)

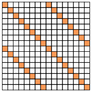
$\mathrm{supp}(\mathbf{X}^{(3)}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(3)}$

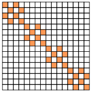
$\mathrm{supp}(\mathbf{X}^{(4)}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(4)}$

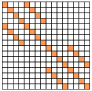
Corresponding rank-one supports are pairwise disjoint.

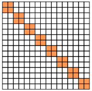
Hierarchical factorization algorithm

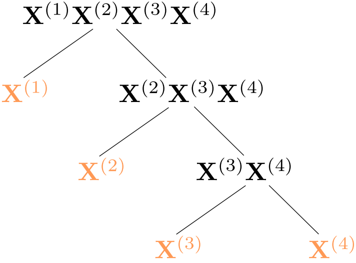
Let $\mathbf{Z} := \mathbf{X}^{(1)}\mathbf{X}^{(2)}\mathbf{X}^{(3)}\mathbf{X}^{(4)}$ such that:

$\mathrm{supp}(\mathbf{X}^{(1)}) \subseteq$ 

$\mathrm{supp}(\mathbf{X}^{(3)}) \subseteq$ 

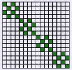
$\mathrm{supp}(\mathbf{X}^{(2)}) \subseteq$ 

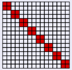
$\mathrm{supp}(\mathbf{X}^{(4)}) \subseteq$ 



The two-layer procedure is repeated **recursively**.

Lemma (Support of the partial products)

$\mathrm{supp}(\mathbf{X}^{(3)}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(3)}$

$\mathrm{supp}(\mathbf{X}^{(4)}) \subseteq$  $= \mathbf{S}_{\mathtt{bf}}^{(4)}$

Corresponding rank-one supports are pairwise disjoint.

$\{\mathbf{X}^{(\ell)}\}_{\ell=1}^{4}$ are recovered from $\mathbf{Z}$, up to scaling ambiguities.

Uniqueness of butterfly factorization

Theorem ([Zheng et al. 2023])

Except for trivial degeneracies, the butterfly factorization $\mathbf{Z} := \mathbf{X}^{(1)} \dots \mathbf{X}^{(J)}$ *is essentially unique, up to unavoidable scaling ambiguities:*

$$\begin{cases} \bar{\mathbf{X}}^{(1)} \dots \bar{\mathbf{X}}^{(J)} = \mathbf{Z} \\ \forall \ell \in [J], \ \mathrm{supp}(\bar{\mathbf{X}}^{(\ell)}) \subseteq \mathbf{S}_{\mathrm{bf}}^{(\ell)} \end{cases} \quad \Longrightarrow \ (\bar{\mathbf{X}}^{(\ell)})_{\ell=1}^{J} \sim (\mathbf{X}^{(\ell)})_{\ell=1}^{J}$$

The unique factors are recovered from the hierarchical algorithm.

Uniqueness of butterfly factorization

Theorem ([Zheng et al. 2023])

Except for trivial degeneracies, the butterfly factorization $\mathbf{Z} := \mathbf{X}^{(1)} \dots \mathbf{X}^{(J)}$ *is essentially unique, up to unavoidable scaling ambiguities:*

$$\begin{cases} \bar{\mathbf{X}}^{(1)} \dots \bar{\mathbf{X}}^{(J)} = \mathbf{Z} \\ \forall \ell \in [J], \operatorname{supp}(\bar{\mathbf{X}}^{(\ell)}) \subseteq \mathbf{S}_{\mathrm{bf}}^{(\ell)} \end{cases} \implies (\bar{\mathbf{X}}^{(\ell)})_{\ell=1}^{J} \sim (\mathbf{X}^{(\ell)})_{\ell=1}^{J}$$

The unique factors are recovered from the hierarchical algorithm.

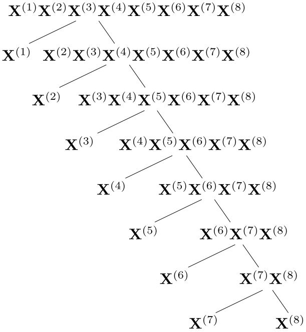
Proof: at level ℓ, the intermediate matrix to factorize is

$$\mathbf{M} := \underbrace{(\mathbf{D}^{-1}\mathbf{X}^{(\ell)})}_{\mathbf{A}} \underbrace{(\mathbf{X}^{(\ell+1)} \dots \mathbf{X}^{(J)} \tilde{\mathbf{D}})}_{\mathbf{B}}.$$

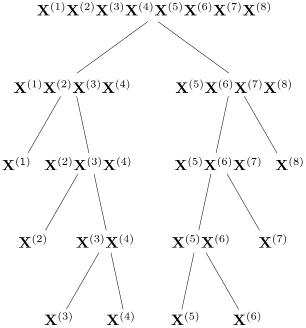
The algorithm recovers $(\mathbf{A}, \mathbf{B})$ because of optimality & uniqueness in $(\star)$.

Flexible choice in the hierarchical order

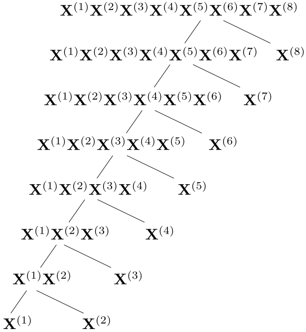
The algorithm works for any factor-bracketing binary tree.



Column-wise factorization



Hybrid factorization



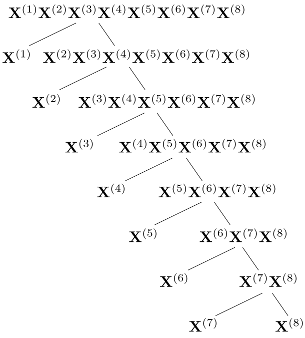
Row-wise factorization

This extends existing work that consider only 3 trees.

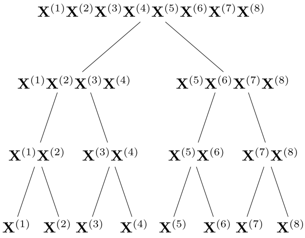
[Y. Liu et al., **Butterfly factorization via randomized matrix-vector multiplications**, SISC, 2021]

Flexible choice in the hierarchical order

The algorithm works for any factor-bracketing binary tree.



Unbalanced tree



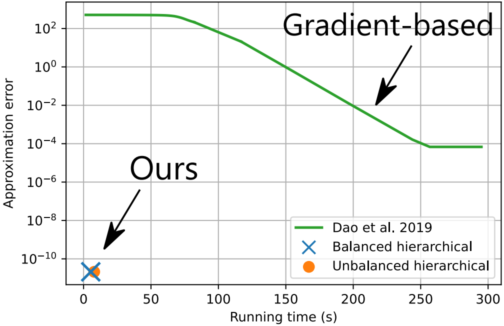
Balanced tree (new)

This extends existing work that consider only 3 trees.

[Y. Liu et al., **Butterfly factorization via randomized matrix-vector multiplications**, SISC, 2021]

Faster and more accurate in the noiseless setting

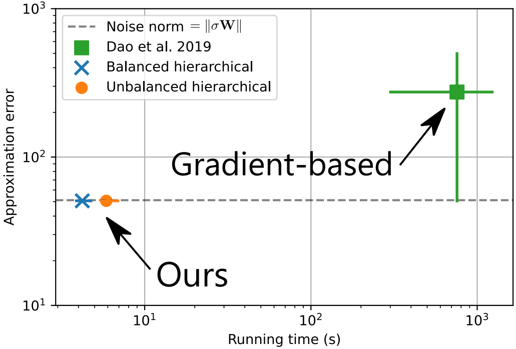
Approximate $\mathbf{Z} := \mathbf{DFT}_{512}$ by a product of $J = 9$ butterfly factors:



The theoretical complexity of the algorithm is $\mathcal{O}(N^2)$.

Also more robust in the noisy setting

Approximate $\mathbf{Z} := \mathbf{DFT}_{512} + \sigma \mathbf{W}$ by a product of $J = 9$ butterfly factors:



FAµST Toolbox: faust.inria.fr

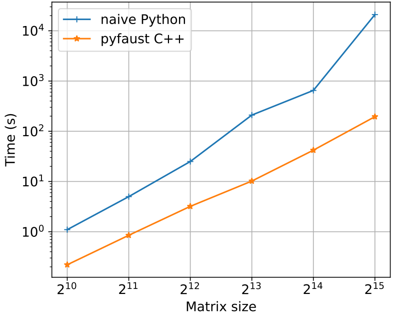
Efficient implem. of fast transforms and algorithms for sparse matrix fact.

- Python & MATLAB wrappers (C++ core, GPU compatible)
- PYPI install: `pip install pyfaust`

Ex: butterfly factorization

```
from pyfaust.fact import butterfly
from pyfaust import Faust, wht

H = wht(8).toarray()
F = butterfly(H, type='bbtree')
(F-H).norm() / Faust(H).norm()
# Output: 1.2560739454502295e-15
```



Factorization of the Hadamard matrix with a balanced tree.

Conclusion and perspectives

1. The butterfly structure captures many common fast transforms.
2. We proved the essential uniqueness of the butterfly factorization.
3. Butterfly factors are recovered by a flexible hierarchical algorithm.

Hierarchical algorithm: $\mathcal{O}(N^2)$

$\mathbf{Z} \in \mathbb{R}^{N \times N}$ (dense)	$\tilde{\mathbf{Z}} := \mathbf{X}^{(J)}\mathbf{X}^{(J-1)} \ldots \mathbf{X}^{(1)}$
Storage: $\mathcal{O}(N^2)$	Storage: $\mathcal{O}(N \log N)$
Cost for evaluation: $\mathcal{O}(N^2)$ $\mathbf{x} \mapsto \mathbf{Z}\mathbf{x}$	Cost for evaluation: $\mathcal{O}(N \log N)$ $\mathbf{x} \mapsto \tilde{\mathbf{Z}}\mathbf{x}$

Conclusion and perspectives

1. The butterfly structure captures many common fast transforms.
2. We proved the essential uniqueness of the butterfly factorization.
3. Butterfly factors are recovered by a flexible hierarchical algorithm.

Hierarchical algorithm: $\mathcal{O}(N^2)$

$\mathbf{Z} \in \mathbb{R}^{N \times N}$ (dense)	$\tilde{\mathbf{Z}} := \mathbf{X}^{(J)}\mathbf{X}^{(J-1)} \dots \mathbf{X}^{(1)}$
Storage: $\mathcal{O}(N^2)$	Storage: $\mathcal{O}(N \log N)$
Cost for evaluation: $\mathcal{O}(N^2)$ $\mathbf{x} \mapsto \mathbf{Z}\mathbf{x}$	Cost for evaluation: $\mathcal{O}(N \log N)$ $\mathbf{x} \mapsto \tilde{\mathbf{Z}}\mathbf{x}$

On going work

- Approximation error of the hierarchical algorithm
- Taking into account row and column permutations
- Efficient training of neural networks with butterfly structure

Thank you for your attention!

To know more:

Q.-T. Le, L. Zheng, E. Riccietti, and R. Gribonval
Fast learning of fast transforms, with guarantees
In *ICASSP*, 2022.

Q.-T. Le, E. Riccietti, and R. Gribonval
Spurious valleys, NP-hardness, and tractability of sparse matrix factorization with fixed support
In *SIAM Journal on Matrix Analysis and Applications*, 2022.

L. Zheng, E. Riccietti, and R. Gribonval
Efficient identification of butterfly sparse matrix factorizations
In *SIAM Journal on Mathematics of Data Science*, 2023.