Paper ID: 4890

Multiplication-Avoiding Variant of Power Iteration with Applications

Hongyi Pan, Diaa Badawi, Runxuan Miao, Erdem Koyuncu, Ahmet Enis Cetin

Electrical and Computer Engineering, University of Illinois Chicago, Chicago, USA

Multiplication-Avoiding Vector Product (MAVP)

Let x and y are two real numbers, $\mathbf{x}$ and $\mathbf{y}$ are two vectors:

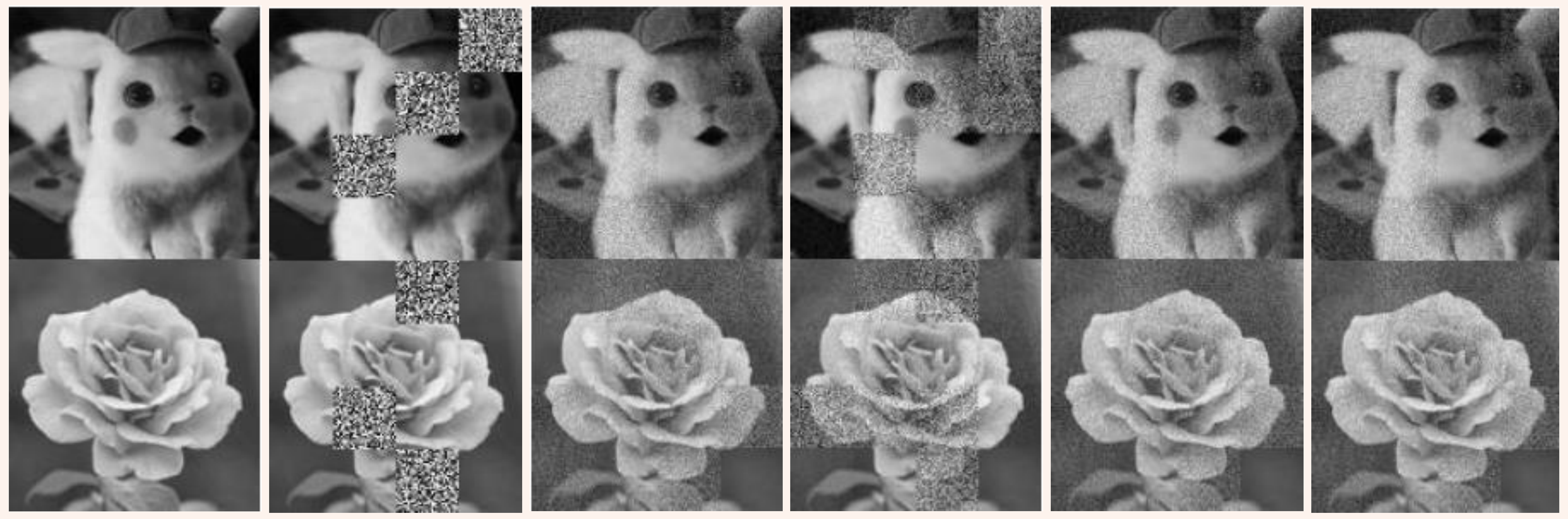
- Min-1 operation:
  - $x \odot y = \text{sign}(x \cdot y)\min(|x|, |y|)$
  - $\mathbf{x}^T \odot \mathbf{y} = \sum_i \text{sign}(x_i \cdot y_i)\min(|x_i|, |y_i|)$
- Min-2 operation:
  - $x \odot_{\text{m}} y = \begin{cases} \min(|x|, |y|), & \text{if sign}(x) = \text{sign}(y) \\ 0, & \text{otherwise} \end{cases}$
  - $\mathbf{x}^T \odot_{\text{m}} \mathbf{y} = \sum_i \mathbf{1}(\text{sign}(x_i) = \text{sign}(y_i))\min(|x_i|, |y_i|)$
- Mercer-Type kernels
  - $< \phi(\mathbf{x}), \phi(\mathbf{y}) >= \sum_i \text{sign}(x_i \cdot y_i)\min(|x_i|, |y_i|)$
  - $< \phi_m(\mathbf{x}), \phi_m(\mathbf{y}) >= \sum_i \mathbf{1}(\text{sign}(x_i) = \text{sign}(y_i))\min(|x_i|, |y_i|)$
- They hold ℓ_1-norm property: $\mathbf{x}^T \odot \mathbf{x} = \mathbf{x}^T \odot_{\mathbf{m}} \mathbf{x} = ||\mathbf{x}||_1$
- They are multiplication-free because the only "multiplication" is from the sign operation, which can be implemented using bit operations.
- Matrix form: Let $\mathbf{X} = [\mathbf{x}_1 \quad \cdots \quad \mathbf{x}_m]$ and $\mathbf{Y} = [\mathbf{y}_1 \quad \cdots \quad \mathbf{y}_p]$,
  - $\mathbf{X}^T \oplus \mathbf{Y} = \begin{bmatrix} \mathbf{x}_1^T \oplus \mathbf{y}_1 & \cdots & \mathbf{x}_1^T \oplus \mathbf{y}_p \\ \vdots & \ddots & \vdots \\ \mathbf{x}_m^T \oplus \mathbf{y}_1 & \cdots & \mathbf{x}_m^T \oplus \mathbf{y}_p \end{bmatrix}$, where $\oplus \in \{\odot, \odot_{\text{m}}\}$.
- Min-1 and Min-2 Covariance matrices:
  - $\mathbf{A} = \frac{1}{N-1}\mathbf{X} \oplus \mathbf{X}^T$, where $\oplus \in \{\odot, \odot_{\text{m}}\}$.

Multiplication-Avoiding Power Iteration (MAPI)

- Regular Power Iteration (RPI):
  - $\mathbf{w}_{t+1} = \frac{\mathbf{A}\mathbf{w}_t}{||\mathbf{A}\mathbf{w}_t||}$
- Multiplication-Avoiding Power Iteration (MAPI):
  - $\mathbf{w}_{t+1} = \frac{\mathbf{A} \oplus \mathbf{w}_t}{||\mathbf{A} \oplus \mathbf{w}_t||_1}$, where $\oplus \in \{\odot, \odot_{\text{m}}\}$.
- $\mathbf{A}$: a diagonalizable matrix, for example, a covariance matrix.
- $\mathbf{w}_t$: the result vector at t-th power iteration, initialized randomly.
- RPI converges to the dominant eigenvector.
- MAPI converges to a "pseudo-eigenvector" in our simulation studies.
- In MAPI, we do ℓ_1-normalization because the two MAVPs hold ℓ_1-norm Property. We can normalize the result vector by its ℓ_2-norm after the final iteration because the ℓ_2-norm of an eigenvector is usually 1.

Experiment 1: Image Reconstruction

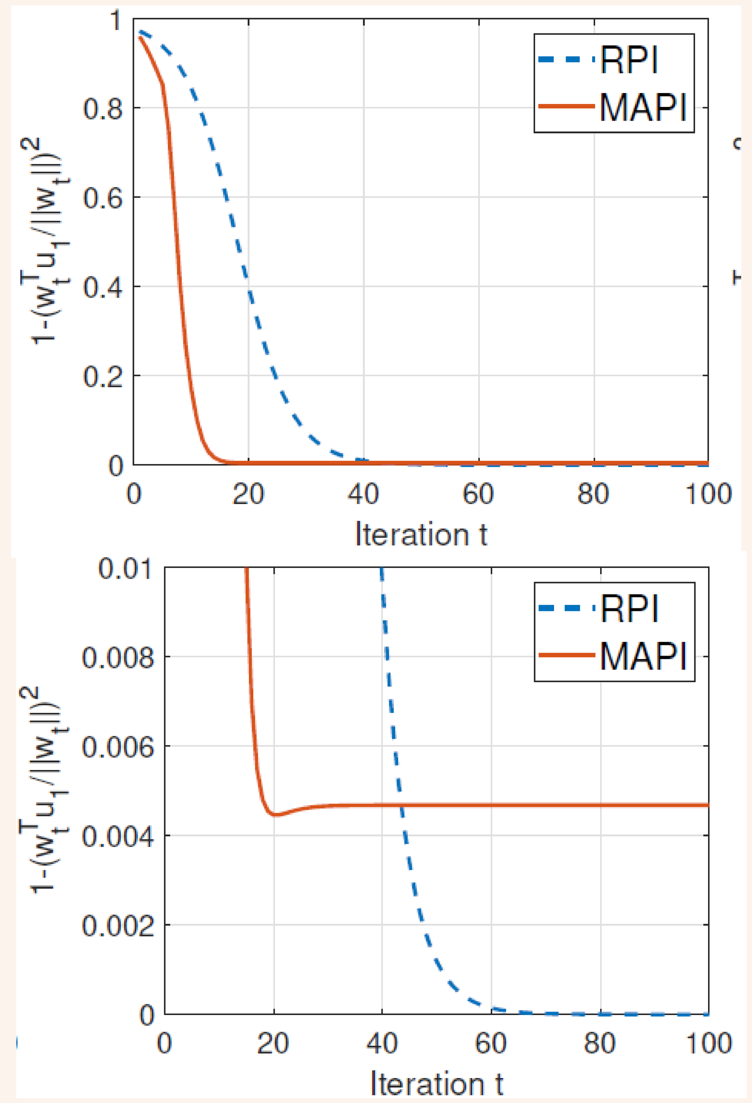
- In each image experiment, we have 10 different occluded images (like (b) shows): $\{X_1, X_2, \dots, X_{10}\}$. All image sizes are 128x128. The occluded images are obtained by adding random noise patches on to the same original image. Suppose the original image (a) is not available, and we want to reconstruct (a). (c)—(f) shows the reconstructed results from different methods.
- We compare our two power iterations (min1-PI and min2-PI) with the recursive ℓ_1-PI and the regular power iteration (RPI) on 14 experiments.
- Our methods are as robust as ℓ1-PCA based restoration, and they overbeat the RPI obviously.



(a) (b) (c) (d) (e) (f)

(a) original images; (b) occluded images; (c) recursive ℓ1-PCA results; (d) RPI results; (e) min1-PI results; (f) min2-PI results.

Experiment 2: Synthetic Dataset



RPI versus MAPI on a synthetic dataset. MAPI does not converge to the same vector as RPI because the equations are different, but the vectors produce the same ranks.

We perform eigen-decomposition and Min1-PI on a synthetic dataset. We plot iteration time t VS $1 - \left(\frac{w_t^T u_1}{||w_t||}\right)^2$. w_t is the result vector from PI, u_1 is the dominant eigenvector.

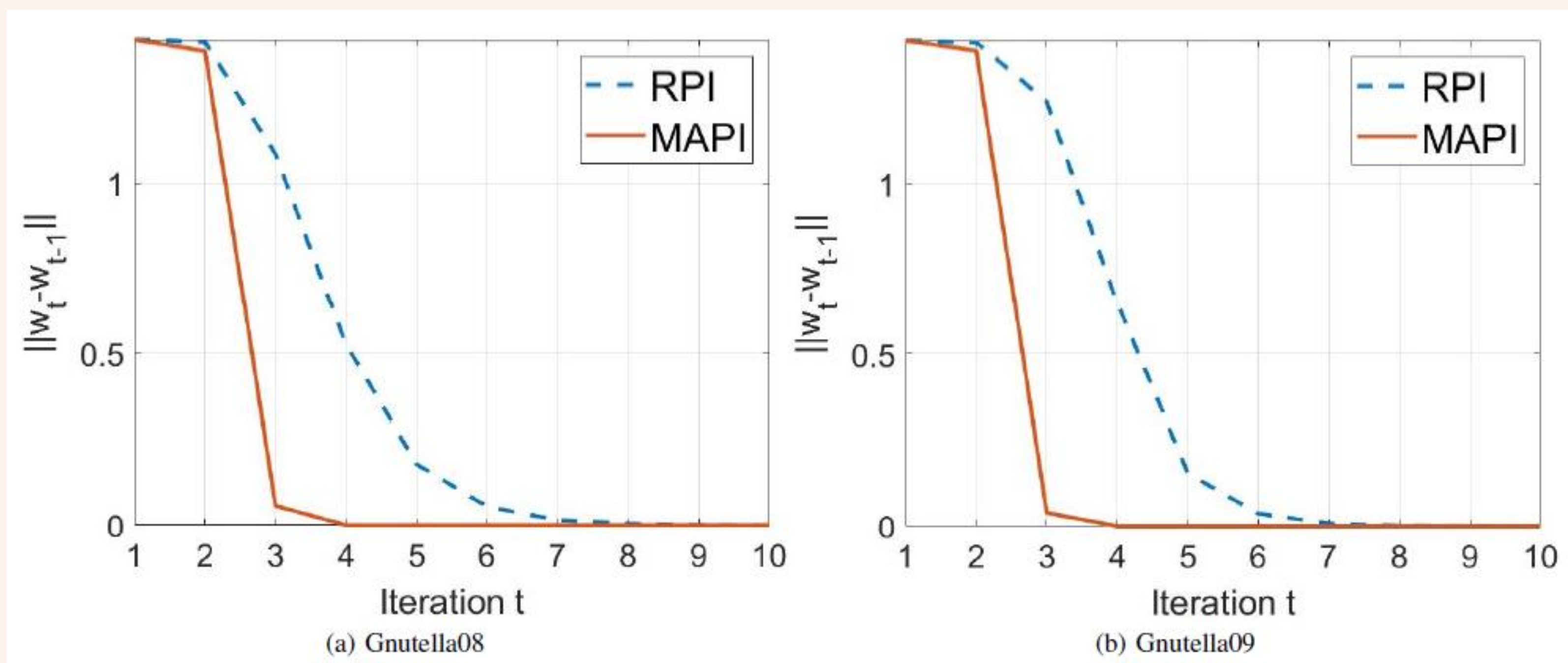
RPI: $w_T = [0.1433, 0.4171, -0.1166, 0.3863, 0.1285, -0.1315, -0.4330, -0.2097, -0.5770, -0.2106]$

MAPI: $w_T = [0.1649, 0.4166, -0.1371, 0.3887, 0.1480, -0.1508, -0.4306, -0.2359, -0.5362, -0.2370]$

When we sort them from the largest to the smallest, **both vectors produce the same ranks** $\{\mathbf{2}, \mathbf{4}, \mathbf{5}, \mathbf{3}, \mathbf{6}, \mathbf{1}, \mathbf{8}, \mathbf{10}, \mathbf{7}, \mathbf{9}\}$.

Multiplication-Avoiding Power Iteration (MAPI)

- The 6,301-node dataset Gnutella08's top-10 ranks of the indices:
  - RPI: 367, 249, 145, 264, 266, 123, 127, 122, 1317, 5;
  - MAPI: 266, 123, 367, 127, 424, 249, 145, 264, 427, 251.
- The 8,114-node dataset Gnutella09's top-10 ranks of the indices:
  - RPI: 351, 563, 822, 534, 565, 825, 1389, 1126, 356, 530,
  - MAPI: 351, 822, 51, 1389, 563, 565, 530, 825, 356, 1074.
- Gnutella08's has 7 common top-10 ranks of indices: 367, 249, 145, 265, 226, 123, 127
- Gnutella09's has 8 common top-10 ranks of indices: 351, 563, 822, 565, 825, 1389, 356, 530.
- If we use MAPI to replace the RPI in the PageRank algorithm, searching results (what links will be shown on the first page) will be close, but the searching time will be reduced significantly.



PageRank convergence curves on Gnutella