A Further Details on the Experiments

In this section, we provide further details on the numerical experiments reported in Section 5.

A.1 Details of Section 5.1

In Section 5.1, we generate the synthetic datasets according to Model 1 with $N = 10$, $d = 6$, $r_i = 6$, and $n_i = 200$ for each $1 \leq i \leq 10$. Each $\mathbf{D}_i^*$ is an orthogonal matrix with the first $r^g = 3$ columns shared with every other client and the last $r_i^l = 3$ columns unique to themselves. Each $\mathbf{X}_i^*$ is first generated from a Gaussian-Bernoulli distribution where each entry is non-zero with a probability 0.2. Then, $\mathbf{X}_i^*$ is further truncated, where all the entries $(\mathbf{X}_i^*)_{(j,k)}$ with $|\left(\mathbf{X}_i^*\right)_{(j,k)}| < 0.3$ are replaced by $(\mathbf{X}_i^*)_{(j,k)} = 0.3 \times \mathrm{sign}((\mathbf{X}_i^*)_{(j,k)})$.

We use the orthogonal DL algorithm (Algorithm 4) introduced in (Liang et al., 2022, Algorithm 1) as the local DL algorithm for each client. This algorithm is simple to implement and comes equipped with a strong convergence guarantee (see (Liang et al., 2022, Theorem 1)). Here $\mathrm{HT}_\zeta(\cdot)$ denotes the hard-thresholding operator at level ζ, which is defined as:

$$(\mathrm{HT}_\zeta(\mathbf{A}))_{(i,j)} = \begin{cases} \mathbf{A}_{(i,j)} & \text{if} \quad |\mathbf{A}_{(i,j)}| \geq \zeta, \\ 0 & \text{if} \quad |\mathbf{A}_{(i,j)}| < \zeta. \end{cases}$$

Specifically, we use $\zeta = 0.15$ for the experiments in Section 5.1. $\mathrm{Polar}(\cdot)$ denotes the polar decomposition operater, which is defined as $\mathrm{Polar}(\mathbf{A}) = \mathbf{U}_\mathbf{A}\mathbf{V}_\mathbf{A}^\top$, where $\mathbf{U}_\mathbf{A}\mathbf{\Sigma}_\mathbf{A}\mathbf{V}_\mathbf{A}^\top$ is the Singular Value Decomposition (SVD) of $\mathbf{A}$.

Algorithm 4 Alternating minimization for orthogonal dictionary learning (Liang et al. (2022))

1: **Input:** $\mathbf{Y}_i$, $\mathbf{D}_i^{(t)}$
2: Set $\mathbf{X}_i^{(t)} = \mathrm{HT}_\zeta\left(\mathbf{D}^{(t)_i\top}\mathbf{Y}_i\right)$
3: Set $\mathbf{D}_i^{(t+1)} = \mathrm{Polar}\left(\mathbf{Y}_i\mathbf{X}_i^{(t)\top}\right)$
4: **return** $\mathbf{D}_i^{(t+1)}$

For a fair comparison, we initialize both strategies using the same $\{\mathbf{D}_i^{(0)}\}_{i=1}^N$, which is obtained by iteratively calling Algorithm 4 with a random initial dictionary and shrinking thresholds ζ. For a detailed discussion on such an initialization scheme we refer the reader to Liang et al. (2022).

A.2 Details of Section 5.2

In section 5.2, we aim to learn a dictionary with imbalanced data collected from MNIST dataset (LeCun et al., 2010). Specifically, we consider $N = 10$ clients, each with 500 handwritten images. Each image is comprised of 28×28 pixels. Instead of randomly assigning images, we construct dataset i such that it contains 450 images of digit i and 50 images of other digits. Here client 10 corresponds to digit 0. After vectorizing each image into a 784×1 one-dimension signal, our imbalanced dataset contains 10 matrices $\mathbf{Y}_i \in \mathbb{R}^{784\times 500}, i = 1, \ldots, 10$.

We first use Algorithm 4 to learn an orthogonal dictionary for each client, using their own imbalanced dataset. For client i, given the output of Algorithm 4 after T iterations $\mathbf{D}_i^{(T)}$, we reconstruct a new signal $\mathbf{y}$ using the top k atoms according to the following steps: first, we solve a *sparse coding* problem to find the sparse code $\mathbf{x}$ such that $\mathbf{y} \approx \mathbf{D}_i^{(T)}\mathbf{x}$. This can be achieved by Step 2 in Algorithm 4. Second, we find the top k entries in $\mathbf{x}$ that have the largest magnitude: $\mathbf{x}_{(\alpha_1,1)}$, $\mathbf{x}_{(\alpha_2,1)}, \cdots, \mathbf{x}_{(\alpha_k,1)}$. Finally, we calculate the reconstructed signal $\tilde{\mathbf{y}}$ as

$$\tilde{\mathbf{y}} = \sum_{j=1}^{k} \mathbf{x}_{(\alpha_h,1)}\left(\mathbf{D}_i^{(T)}\right)_{\alpha_h}.$$

The second row of Figure 3 is generated by the above procedure with $k = 5$ using the dictionary learned by Client 1. The third row of Figure 3 corresponds to the reconstructed images using the output of PerMA.

A.3 Details of Section 5.3

Our considered dataset in section 5.3 contains 62 frames, each of which is a $480 \times 640 \times 3$ RGB image. We consider each frame as one client ($N = 62$). After dividing each frame into 40×40 patches, we obtain each data matrix $\mathbf{Y}_i \in \mathbb{R}^{576\times 1600}$. Then we apply PerMA to $\{\mathbf{Y}_i\}_{i=1}^{62}$ with $r_i = 576$ for all i and $r^g = 30$. Consider $\mathbf{D}_i^{(T)} = \begin{bmatrix}\mathbf{D}^{g,(T)} & \mathbf{D}_i^{l,(T)}\end{bmatrix}$, which is the output of PerMA for client i. We reconstruct each $\mathbf{Y}_i$ using the procedure described in the previous section with $k = 50$. Specifically, we separate the contribution of $\mathbf{D}^{g,(T)}$ from $\mathbf{D}_i^{l,(T)}$. Consider the reconstructed matrix $\tilde{Y}_i$ as

$$\tilde{\mathbf{Y}}_i = \begin{bmatrix}\mathbf{D}^{g,(T)} & \mathbf{D}_i^{l,(T)}\end{bmatrix}\begin{bmatrix}\mathbf{X}_i^g \\ \mathbf{X}_i^l\end{bmatrix} = \underbrace{\mathbf{D}^{g,(T)}\mathbf{X}_i^g}_{\tilde{\mathbf{Y}}_i^g} + \underbrace{\mathbf{D}_i^{l,(T)}\mathbf{X}_i^l}_{\tilde{\mathbf{Y}}_i^l}$$

The second column and the third column of Figure 4 correspond to reconstructed results of $\tilde{\mathbf{Y}}_i^g$ and $\tilde{\mathbf{Y}}_i^l$ respectively. We can see clear separation of the background (which is shared among all frames) from the moving objects (which is unique to each frame).

One notable difference between this experiment and the previous one is in the choice of the DL algorithm $\mathcal{A}_i$. To provide more flexibility, we relax the orthogonality condition for the dictionary. Therefore, we use the alternating minimization algorithm introduced in Arora et al. (2015) for each client (see Algorithm 5). The main difference between this algorithm and Algorithm 4 is that the polar decomposition step in Algorithm 4 is replaced by a single iteration of the gradient descent applied to the loss function $\mathcal{L}(\mathbf{D}, \mathbf{X}) = \|\mathbf{D}\mathbf{X} - \mathbf{Y}\|_F^2$.

Algorithm 5 Alternating minimization for general dictionary learning (Arora et al. (2015))

1: **Input:** $\mathbf{Y}_i$, $\mathbf{D}_i^{(t)}$
2: Set $\mathbf{X}_i^{(t)} = \mathrm{HT}_\zeta\left(\mathbf{D}_i^{(t)\top}\mathbf{Y}_i\right)$
3: Set $\mathbf{D}_i^{(t+1)} = \mathbf{D}_i^{(t)} - 2\eta\left(\mathbf{D}_i^{(t)}\mathbf{X}_i^{(t)} - \mathbf{Y}_i\right)\mathbf{X}_i^{(t)\top}$
4: **return** $\mathbf{D}_i^{(t+1)}$

Even with the computational saving brought up by Algorithm 5, the runtime significantly slows down for PerMA due to large N, d, and p. Here we report a practical trick to speed up PerMA, which is a local refinement procedure (Algorithm 6) added immediately before `local_update` (Step 10 of Algorithm 1). At a high level, `local_dictionary_refinement` first finds the local residual data matrix $\mathbf{Y}_i^l$ by removing the contribution of the global dictionary. Then it iteratively refines the local dictionary with respect to $\mathbf{Y}_i^l$. We observed that `local_dictionary_refinement` significantly improves the local reconstruction quality. We leave its theoretical analysis as a possible direction for future work.

Algorithm 6 `local_dictionary_refinement`

1: **Input:** $\mathbf{D}_i^{(t)} = \begin{bmatrix}\mathbf{D}^{g,(t)} & \mathbf{D}_i^{l,(t)}\end{bmatrix}$, $\mathbf{Y}_i$
2: Find $\begin{bmatrix}\mathbf{X}_i^g \\ \mathbf{X}_i^l\end{bmatrix}$ such that $\mathbf{Y}_i \approx \begin{bmatrix}\mathbf{D}^{g,(t)} & \mathbf{D}_i^{l,(t)}\end{bmatrix}\begin{bmatrix}\mathbf{X}_i^g \\ \mathbf{X}_i^l\end{bmatrix}$ `// Solving a sparse coding problem`
3: Set $\mathbf{Y}_i^l = \mathbf{Y}_i - \mathbf{D}^{g,(t)}\mathbf{X}_i^g$
4: Set $\mathbf{D}_i^{\text{refine},(0)} = \mathbf{D}_i^{l,(t)}$.
5: **for** $\tau = 0, 1, ..., T^{\text{refine}} - 1$ **do**
6: Set $\mathbf{D}_i^{\text{refine},(\tau+1)} = \mathcal{A}_i\left(\mathbf{Y}_i^l, \mathbf{D}_i^{\text{refine},(\tau)}\right)$ `// Improving local dictionary`
7: **end for**
8: **return** $\mathbf{D}_i^{\text{refine},(T^{\text{refine}})}$ as refined $\mathbf{D}_i^{l,(t)}$

B Further Discussion on Linearly Convergent Algorithms

In this section, we discuss a linearly convergent DL algorithm that satisfies the conditions of our Theorem 2. In particular, the next theorem is adapted from (Arora et al., 2015, Theorem 12) and shows that a modified variant of Algorithm 5 introduced in (Arora et al., 2015, Algorithm 5) is indeed linearly-convergent.

Theorem 3 (Linear convergence of Algorithm 5 in Arora et al. (2015))**.** *Suppose that the data matrix satisfies $\mathbf{Y} = \mathbf{D}^*\mathbf{X}^*$, where $\mathbf{D}^*$ is an μ-incoherent dictionary and the sparse code $\mathbf{X}^*$ satisfies the generative model introduced in Section 1.2 and Section 4.1 of Arora et al. (2015). For any initial dictionary $\|\mathbf{D}^{(0)}\|_2 \leq 1$, Algorithm 5 in Arora et al. (2015) is (δ, ρ, ψ)-linearly convergent with $\delta = O(1/\log d)$, $\rho \in (1/2, 1)$, and $\psi = O(d^{-\omega(1)})$.*

Algorithm 5 in Arora et al. (2015) is a refinement of Algorithm 5, where the error is further reduced by projecting out the components along the column currently being updated. For brevity, we do not discuss the exact implementation of the algorithm; an interested reader may refer to Arora et al. (2015) for more details. Indeed, we have observed in our experiments that the additional projection step does not provide a significant benefit over Algorithm 5.

C Proof of Theorems

C.1 Proof of Theorem 1

To begin with, we establish a triangular inequality for $d_{1,2}(\cdot,\cdot)$, which will be important in our subsequent arguments:

Lemma 1 (Triangular inequality for $d_{1,2}(\cdot,\cdot)$)**.** *For any dictionary $\mathbf{D}_1$, $\mathbf{D}_2$, $\mathbf{D}_3 \in \mathbb{R}^{d\times r}$, we have*

$$d_{1,2}\left(\mathbf{D}_1, \mathbf{D}_2\right) \leq d_{1,2}\left(\mathbf{D}_1, \mathbf{D}_3\right) + d_{1,2}\left(\mathbf{D}_3, \mathbf{D}_2\right) \tag{13}$$

Proof. Suppose $\mathbf{\Pi}_{1,3}$ and $\mathbf{\Pi}_{3,2}$ satisfy $d_{1,2}\left(\mathbf{D}_1, \mathbf{D}_3\right) = \|\mathbf{D}_1\mathbf{\Pi}_{1,3} - \mathbf{D}_3\|_{1,2}$ and $d_{1,2}\left(\mathbf{D}_3, \mathbf{D}_2\right) = \|\mathbf{D}_3 - \mathbf{D}_2\mathbf{\Pi}_{3,2}\|_{1,2}$. Then we have

$$\begin{aligned} d_{1,2}\left(\mathbf{D}_1, \mathbf{D}_3\right) + d_{1,2}\left(\mathbf{D}_3, \mathbf{D}_2\right) &= \|\mathbf{D}_1\mathbf{\Pi}_{1,3} - \mathbf{D}_3\|_{1,2} + \|\mathbf{D}_3 - \mathbf{D}_2\mathbf{\Pi}_{3,2}\|_{1,2} \\ &\geq \|\mathbf{D}_1\mathbf{\Pi}_{1,3} - \mathbf{D}_2\mathbf{\Pi}_{3,2}\|_{1,2} \\ &\geq d_{1,2}\left(\mathbf{D}_1, \mathbf{D}_2\right). \end{aligned} \tag{14}$$

□

Given how the directed graph $\mathcal{G}$ is constructed and modified, any directed path from s to t will be of the form $\mathcal{P} = s \to (\mathbf{D}_1^{(0)})_{\alpha(1)} \to (\mathbf{D}_2^{(0)})_{\alpha(2)} \to \cdots \to (\mathbf{D}_N^{(0)})_{\alpha(N)} \to t$. Specifically, each layer (or client) contributes exactly one node (or atom), and the path is determined by $\alpha(\cdot) : [N] \to [r]$. Recall that $\mathbf{D}_i^* = \begin{bmatrix}\mathbf{D}^{g*} & \mathbf{D}_i^{l*}\end{bmatrix}$ for every $1 \leq i \leq N$. Assume, without loss of generality, that for every client $1 \leq i \leq N$,

$$\mathbf{I}_{r_i\times r_i} = \arg\min_{\mathbf{\Pi}\in\mathcal{P}(r_i)} \left\|\mathbf{D}_i^*\mathbf{\Pi} - \mathbf{D}_i^{(0)}\right\|_{1,2}. \tag{15}$$

In other words, the first r^g atoms in the initial dictionaries $\{D_i^{(0)}\}_{i=1}^N$ are aligned with the global dictionary. Now consider the special path $\mathcal{P}_j^*$ for $1 \leq j \leq r^g$ defined as

$$\mathcal{P}_j^* = s \to (\mathbf{D}_1^{(0)})_j \to (\mathbf{D}_2^{(0)})_j \to \cdots \to (\mathbf{D}_N^{(0)})_j \to t. \tag{16}$$

To prove that Algorithm 2 correctly selects and aligns global atoms from clients, it suffices to show that $\{\mathcal{P}_j^*\}_{j=1}^{r^g}$ are the top-r^g shortest paths from s to t in $\mathcal{G}$. The length of the path $\mathcal{P}_j^*$ can be bounded as

$$
\begin{aligned}
\mathcal{L}\left(\mathcal{P}_j^*\right) &= \sum_{i=1}^{N-1} d_2\left((\mathbf{D}_i^{(0)})_j, (\mathbf{D}_{i+1}^{(0)})_j\right) \\
&= \sum_{i=1}^{N-1} \min\left\{\|(\mathbf{D}_i^{(0)})_j - (\mathbf{D}_{i+1}^{(0)})_j\|_2, \|(\mathbf{D}_i^{(0)})_j + (\mathbf{D}_{i+1}^{(0)})_j\|_2\right\} \\
&\le \sum_{i=1}^{N-1} \|(\mathbf{D}_i^{(0)})_j - (\mathbf{D}_{i+1}^{(0)})_j\|_2 \\
&\le \sum_{i=1}^{N-1} \|(\mathbf{D}_i^{(0)})_j - (\mathbf{D}^{g*})_j\|_2 + \|(\mathbf{D}_{i+1}^{(0)})_j - (\mathbf{D}^{g*})_j\|_2 \\
&\le \sum_{i=1}^{N-1} (\epsilon_i + \epsilon_{i+1}) \\
&\le 2\sum_{i=1}^{N} \epsilon_i.
\end{aligned}
\tag{17}
$$

We move on to prove that all the other paths from s to t will have a distance longer than $2\sum_{i=1}^{N}\epsilon_i$. Consider a general directed path $\mathcal{P} = s \to (\mathbf{D}_1^{(0)})_{\alpha(1)} \to (\mathbf{D}_2^{(0)})_{\alpha(2)} \to \cdots \to (\mathbf{D}_N^{(0)})_{\alpha(N)} \to t$ that is not in $\{\mathcal{P}_j^*\}_{j=1}^{r^g}$. Based on whether or not $\mathcal{P}$ contains atoms that align with the true global ground atoms, there are two situations:

Case 1: Suppose there exists $1 \le i \le N$ such that $\alpha(i) \le r^g$. Given Model 1 and the assumed equality (15), we know that for layer i, $\mathcal{P}$ contains a global atom. Since $\mathcal{P}$ is not in $\{\mathcal{P}_j^*\}_{j=1}^{r^g}$, there must exist $k \ne i$ such that $\alpha(k) \ne \alpha(i)$. As a result, we have

$$
\begin{aligned}
\mathcal{L}(\mathcal{P}) &\overset{(a)}{\ge} d_{1,2}\left((\mathbf{D}_i^{(0)})_{\alpha(i)}, (\mathbf{D}_k^{(0)})_{\alpha(k)}\right) \\
&\overset{(b)}{\ge} \min\left\{\|(\mathbf{D}_i^*)_{\alpha(i)} - (\mathbf{D}_k^*)_{\alpha(k)}\|_2, (\mathbf{D}_i^*)_{\alpha(i)} + (\mathbf{D}_k^*)_{\alpha(k)}\|_2\right\} \\
&\quad - \|(\mathbf{D}_i^*)_{\alpha(i)} - (\mathbf{D}_i^{(0)})_{\alpha(i)}\|_2 - \|(\mathbf{D}_k^*)_{\alpha(k)} - (\mathbf{D}_k^{(0)})_{\alpha(k)}\|_2 \\
&\overset{(c)}{\ge} \sqrt{2 - 2\left|\left\langle(\mathbf{D}_k^*)_{\alpha(i)}, (\mathbf{D}_k^*)_{\alpha(k)}\right\rangle\right|} - 2\max_{1\le i\le N}\epsilon_i \\
&\overset{(d)}{\ge} \sqrt{2 - 2\frac{\mu}{\sqrt{d}}} - 2\max_{1\le i\le N}\epsilon_i \\
&\overset{(e)}{\ge} 2\sum_{i=1}^{N}\epsilon_i^g
\end{aligned}
\tag{18}
$$

Here (a) and (b) are due to Lemma 1, (c) is due to assumed equality (15), (d) is due to the μ-incoherency of $\mathbf{D}_k^*$, and finally (e) is given by the assumption of Theorem 1.

Case 2: Suppose $\alpha(i) > r^g$ for all $1 \le i \le N$, which means that the path $\mathcal{P}$ only uses approximations to local atoms. Consider the ground truth of these approximations, $(\mathbf{D}_1^*)_{\alpha(1)}, (\mathbf{D}_2^*)_{\alpha(2)}, ..., (\mathbf{D}_N^*)_{\alpha(N)}$. There must exist $1 \le i, j \le N$ such that $d_{1,2}\left((\mathbf{D}_i^*)_{\alpha(i)}, (\mathbf{D}_j^*)_{\alpha(j)}\right) \ge \beta$. Otherwise, it is easy to see that $\{\mathbf{D}_i^{l*}\}_{i=1}^{N}$ would not be β-identifiable because any $(\mathbf{D}_i^*)_{\alpha(i)}$ will satisfy (6). As a result, we have the following:

$$\begin{aligned}
\mathcal{L}(\mathcal{P}) &\geq d_{1,2}\left((\mathbf{D}_i^{(0)})_{\alpha(i)}, (\mathbf{D}_j^{(0)})_{\alpha(j)}\right) \\
&\geq d_{1,2}\left((\mathbf{D}_i^*)_{\alpha(i)}, (\mathbf{D}_j^*)_{\alpha(j)}\right) - \|(\mathbf{D}_i^*)_{\alpha(i)} - (\mathbf{D}_i^{(0)})_{\alpha(i)}\|_2 - \|(\mathbf{D}_j^*)_{\alpha(j)} - (\mathbf{D}_j^{(0)})_{\alpha(j)}\|_2 \\
&\geq \beta - 2\max_i \epsilon_i \\
&\geq 2\sum_{i=1}^N \epsilon_i
\end{aligned} \tag{19}$$

So we have shown that $\{\mathcal{P}_j^*\}_{j=1}^{r^g}$ are the top-r^g shortest paths from s to t in $\mathcal{G}$. Moreover, it is easy to show that $\operatorname{sign}\left(\left\langle(\mathbf{D}_1^{(0)})_j, (\mathbf{D}_i^{(0)})_j\right\rangle\right) = 1$ for small enough $\{\epsilon_i\}_{i=1}^N$. Therefore, the proposed algorithm correctly recovers the global dictionaries (with the correct identity permutation). Finally, we have $\mathbf{D}^{g,(0)} = \frac{1}{N}\sum_{i=1}^N (\mathbf{D}_i^{(0)})_{1:r^g}$, which leads to:

$$\begin{aligned}
d_{1,2}\left(\mathbf{D}^{g,(0)}, \mathbf{D}^{g*}\right) &\leq \max_{1\leq j\leq r^g} \left\|\frac{1}{N}\sum_{i=1}^N (\mathbf{D}_i^{(0)})_j - (\mathbf{D}^{g*})_j\right\|_2 \\
&\leq \max_{1\leq j\leq r^g} \frac{1}{N}\sum_{i=1}^N \left\|(\mathbf{D}_i^{(0)})_j - (\mathbf{D}^{g*})_j\right\|_2 \\
&\leq \max_{1\leq j\leq r^g} \frac{1}{N}\sum_{i=1}^N \epsilon_i \\
&= \frac{1}{N}\sum_{i=1}^N \epsilon_i.
\end{aligned} \tag{20}$$

This completes the proof of Theorem 1. $\square$

C.2 Proof of Theorem 2

Throughout this section, we define:

$$\bar{\rho} := \frac{1}{N}\sum_{i=1}^N \rho_i, \qquad \bar{\psi} := \frac{1}{N}\sum_{i=1}^N \psi_i. \tag{21}$$

We will prove the convergence of the global dictionary in Theorem 2 by proving the following induction: at each $t \geq 1$, we have

$$d_{1,2}\left(\mathbf{D}^{g,(t+1)}, \mathbf{D}^{g*}\right) \leq \bar{\rho} d_{1,2}\left(\mathbf{D}^{g,(t)}, \mathbf{D}^{g*}\right) + \bar{\psi}. \tag{22}$$

At the beginning of communication round t, each client i performs `local_update` to get $\mathbf{D}_i^{(t+1)}$ given $\begin{bmatrix}\mathbf{D}^{g,(t)} & \mathbf{D}_i^{l,(t)}\end{bmatrix}$. Without loss of generality, we assume

$$\mathbf{I}_{r_i\times r_i} = \arg\min_{\mathbf{\Pi}\in\mathcal{P}(r_i)} \left\|\mathbf{D}_i^*\mathbf{\Pi} - \begin{bmatrix}\mathbf{D}^{g,(t)} & \mathbf{D}_i^{l,(t)}\end{bmatrix}\right\|_{1,2}, \tag{23}$$

$$\mathbf{I}_{r_i\times r_i} = \arg\min_{\mathbf{\Pi}\in\mathcal{P}(r_i)} \left\|\mathbf{D}_i^*\mathbf{\Pi} - \mathbf{D}_i^{(t+1)}\right\|_{1,2}. \tag{24}$$

Assumed equalities (23) and (24) imply that the permutation matrix that aligns the input and the output of $\mathcal{A}_i$ is also $\mathbf{I}_{r_i\times r_i}$. Specifically, the linear convergence property of $\mathcal{A}_i$ and Theorem 1 thus suggest:

$$\left\|\left(\mathbf{D}_i^{(t+1)}\right)_j - (\mathbf{D}_i^*)_j\right\|_2 \leq \rho_i \left\|\left(\mathbf{D}^{g,(t)}\right)_j - (\mathbf{D}_i^*)_j\right\|_2 + \psi_i \quad \forall 1\leq j\leq r^g, 1\leq i\leq N. \tag{25}$$

However, our algorithm is unaware of this trivial alignment. We will next show the remaining steps in `local_update` correctly recovers the identity permutation. The proof is very similar to the proof of Theorem 1 since we are essentially running Algorithm 2 on a two-layer $\mathcal{G}$. For every $1 \leq i \leq N$, $1 \leq j \leq r^g$, we have

$$\begin{aligned} d_{1,2}\left(\left(\mathbf{D}_i^{(t+1)}\right)_j, \left(\mathbf{D}^{g,(t)}\right)_j\right) &\leq d_{1,2}\left(\left(\mathbf{D}_i^{(t+1)}\right)_j, \left(\mathbf{D}_i^*\right)_j\right) + d_{1,2}\left(\left(\mathbf{D}_i^*\right)_j, \left(\mathbf{D}^{g,(t)}\right)_j\right) \\ &\leq 2\delta_i. \end{aligned} \tag{26}$$

Meanwhile for $k \neq j$,

$$\begin{aligned} & d_{1,2}\left(\left(\mathbf{D}_i^{(t+1)}\right)_k, \left(\mathbf{D}^{g,(t)}\right)_j\right) \\ \geq\ & d_{1,2}\left(\left(\mathbf{D}_i^*\right)_k, \left(\mathbf{D}_i^*\right)_j\right) - d_{1,2}\left(\left(\mathbf{D}_i^{(t+1)}\right)_k, \left(\mathbf{D}_i^*\right)_k\right) - d_{1,2}\left(\left(\mathbf{D}_i^*\right)_j, \left(\mathbf{D}^{g,(t)}\right)_j\right) \\ \geq & \sqrt{2 - \frac{2\mu}{\sqrt{d}}} - 2\delta_i. \\ \geq & 2\delta_i. \end{aligned} \tag{27}$$

As a result, we successfully recover the identity permutation, which implies

$$\left\|\left(\mathbf{D}_i^{g,(t+1)}\right)_j - \left(\mathbf{D}_i^{g*}\right)_j\right\|_2 \leq \rho_i \left\|\left(\mathbf{D}^{g,(t)}\right)_j - \left(\mathbf{D}_i^{g*}\right)_j\right\|_2 + \psi_i \quad \forall 1 \leq j \leq r^g, 1 \leq i \leq N. \tag{28}$$

Finally, the aggregation step (Step 13 in Algorithm 1) gives:

$$\begin{aligned} d_{1,2}\left(\mathbf{D}^{g,(t+1)}, \mathbf{D}^{g*}\right) &\leq \left\|\frac{1}{N}\sum_{i=1}^N \mathbf{D}_i^{g,(t+1)} - \mathbf{D}^{g*}\right\|_{1,2} \\ &= \max_{1 \leq j \leq r^g} \left\|\left(\frac{1}{N}\sum_{i=1}^N \mathbf{D}_i^{g,(t+1)}\right)_j - \left(\mathbf{D}^{g*}\right)_j\right\| \\ &\leq \max_{1 \leq j \leq r^g} \frac{1}{N}\sum_{i=1}^N \left\|\left(\mathbf{D}_i^{g,(t+1)}\right)_j - \left(\mathbf{D}_i^{g*}\right)_j\right\|_2 \\ &\leq \max_{1 \leq j \leq r^g} \frac{1}{N}\sum_{i=1}^N \left(\rho_i \left\|\left(\mathbf{D}^{g,(t)}\right)_j - \left(\mathbf{D}_i^{g*}\right)_j\right\|_2 + \psi_i\right) \\ &\leq \frac{1}{N}\sum_{i=1}^N \left(\rho_i d_{1,2}\left(\mathbf{D}^{g,(t)}, \mathbf{D}^{g*}\right) + \psi_i\right) \\ &= \bar{\rho} d_{1,2}\left(\mathbf{D}^{g,(t)}, \mathbf{D}^{g*}\right) + \bar{\psi}. \end{aligned} \tag{29}$$

As a result, we prove the induction (22) for all $0 \leq t \leq T-1$. Inequality (12) is a by-product of the accurate separation of global and local atoms and can be proved by similar arguments. The proof is hence complete. $\square$