SUPPLEMENTARY MATERIAL

In this supplementary material, we prove Theorems 1 and 2 in the main paper. Theorem 3 can be proved using almost the same techniques as the proof for Theorem 1 and hence those details are omitted.

A. Proof of Theorem 1

Proof. Since we have mentioned that, under the case of this theorem the matrices P_{ij} satisfy the sum $\sum\limits_{i=1}^{n}\sum\limits_{j=1}^{n} \operatorname{tr}(P_{ij}^\top T_{ij})$ reaches the optimal value for the objective in Equation (2) in the main paper, it is sufficient to show that the matrices $A_1, \ldots, A_n$ returned by Algorithm 1 satisfy $P_{ij} = A_i^\top A_j$ for each P_{ij}. We first show that, before the coordinate update part (line 9 to 12) of Algorithm 1, we have already ensured that the matrices $A_1, \ldots, A_n$ satisfy the property $P_{ij} = A_i^\top A_j$ for each P_{ij}. We will use induction to prove that, after each iteration during the initialization part of the Algorithm 1, for any set S_k and any $v_i, v_j \in S_k$, we have $P_{ij} = A_i^\top A_j$.

1. Initially (after the 0^{th} iteration), each set S_k only contains one vertex v_k. Since $P_{ii} = I = A_k^\top A_k$, the induction assumption is correct.

2. Assume the induction assumption is correct after the t^{th} iteration ($t \geq 0$). For the $(t+1)^{th}$ iteration, denote the edge we use in this iteration as (v_i, v_j). Then, from the algorithm we know that the matrix $\hat{P} = \operatorname*{argmax}_P \operatorname{tr}(P^\top A_i T_{ij} A_j^\top) = A_i P_{ij} A_j^\top$ for the old values of A_i and A_j. Therefore, after the update on line 5, we will get $P_{ij} = A_i^\top A_j$ for the new values of A_i and A_j. Since we are multiplying on the lefthand side the matrices $A_{j'}$'s on line 5 by the same matrix $\hat{P}$, this does not break the induction assumption inside the set S_j. After the update on line 5, for each $v_{i'} \in S_i$ and each $v_{j'} \in S_j$, we have $A_{i'}^\top A_{j'} = A_{i'}^\top A_i A_i^\top A_j A_j^\top A_{j'} = P_{i'i} P_{ij} P_{jj'} = P_{i'j'}$. Hence, after line 6 and 7, we know that for each $v_k \in S'$ (the set defined on the line 6) and each $v_{i'}, v_{j'} \in S_k$, we have $P_{i'j'} = A_{i'}^\top A_{j'}$. Since the permutation matrices that are changed during this iteration have their corresponding vertices in the set S', we know that the induction assumption is correct after this iteration.

From 1, 2 we know that we have $P_{ij} = A_i^\top A_j$ for each P_{ij} after initialization. Since we have shown in the main paper that the Pairwise Alignment method can solve the problem optimally on this case, we know that our algorithm has also solved the problem optimally after initialization, and hence we do not have any updates in the coordinate update part. Therefore, Algorithm 1 guarantees an optimal solution in this case. □

B. Proof of Theorem 2

Proof. Ideally, we want to recover $(A_1, \ldots, A_n)$ such that $A_i^\top A_j = \hat{T}_{ij}$ for each each pair (A_i, A_j). Let us analyze the probability that we recover such a tuple of $(A_1, \ldots, A_n)$ under the model in Equation (5).

First, let us consider the probability that we recover the correct permutation matrices $\hat{T}_{ij}$ from the optimization problem $\max\limits_P \operatorname{tr}(P^\top T_{ij})$ for any $i \neq j$. For any permutation matrix $P' \in \mathcal{P}_m$, $P' \neq T_{ij}$, if we denote k to be the number of entries where T_{ij} equals 1 but P' does not equal 1, then $k = \operatorname{tr}((\hat{T}_{ij} - P')^\top \hat{T}_{ij})$. Therefore, $U := \frac{\operatorname{tr}(P'^\top T_{ij}) - \operatorname{tr}(\hat{T}_{ij}^\top T_{ij}) + k}{\eta_{ij}}$ follows the Chi-Square distribution $\chi^2(2k)$. Hence, the probability that P' is a better permutation matrix compared to $\hat{T}_{ij}$ is

$$\begin{aligned}&\Pr[\operatorname{tr}(P'^\top T_{ij}) \geq \operatorname{tr}(\hat{T}_{ij}^\top T_{ij})] \\ &= \Pr[\eta_{ij} U - k \geq 0] = \Pr[\frac{U}{\mathbb{E}[U]} - 1 \geq \frac{1}{2}(\frac{1}{\eta_{ij}} - 2)].\end{aligned} \tag{7}$$

For $\eta_{ij} \leq \frac{1}{10}$, by the Chi-Square tail bounds that Laurent and Massart (2000) proposed,

$$\begin{aligned}&\Pr[\frac{U_{ij}}{\mathbb{E}[U_{ij}]} - 1 \geq \frac{1}{2}(\frac{1}{\eta_{ij}} - 2)] \\ \leq &\Pr[\frac{U_{ij}}{\mathbb{E}[U_{ij}]} - 1 \geq \frac{1}{4}(\frac{1}{\eta_{ij}} - 2) + \sqrt{\frac{1}{2}(\frac{1}{\eta_{ij}} - 2)}] \\ \leq &\exp(-\frac{k}{4}(\frac{1}{\eta_{ij}} - 2)).\end{aligned} \tag{8}$$

Denote the probability of misaddressing k letters to k envelopes (The Bernoulli-Euler Problem of the Misaddressed Letters (Dörrie, 2013)) as $p_k = \sum\limits_{i=0}^{\infty} \frac{(-1)^i}{i!} \leq \frac{1}{2}$ (for $k \geq 2$). Then, by union bound on Equation (8) for $k = 2, 3, \ldots, n$, we know the probability that some $P' \neq \hat{T}_{ij}$ is better than $\hat{T}_{ij}$ is at most

$$\begin{aligned}&\sum_{k=2}^{m} p_k \cdot \frac{m!}{(m-k)!} \cdot \exp(-\frac{k}{4}(\frac{1}{\eta_{ij}} - 2)) \\ \leq &\frac{1}{2} \sum_{k=2}^{m} m^k \cdot \exp(-\frac{k}{4}(\frac{1}{\eta_{ij}} - 2)).\end{aligned} \tag{9}$$

If we have $\eta_{ij} \leq \frac{1}{4(1+\varepsilon)\ln m+2}$ for some $\varepsilon > 0$, then,

$$\begin{aligned} &\frac{1}{2}\sum_{k=2}^{m} m^k \cdot \exp(-\frac{k}{4}(\frac{1}{\eta_{ij}} - 2)) \\ &\leq \frac{1}{2}\sum_{k=2}^{m} m^{-\varepsilon k} = \frac{m^{-2\varepsilon}}{2(1-m^{-\varepsilon})}. \end{aligned} \tag{10}$$

Hence, if we choose the variance parameter $\eta_{ij} \leq \min(\frac{1}{10}, \frac{1}{4(1+\varepsilon)\ln m+2})$ for some $\varepsilon > 0$, then for $m \geq 2^{\frac{1}{\varepsilon}}$ we have probability at least $1 - m^{-2\varepsilon}$ to guarantee that we recover $\hat{T}_{ij}$ from the optimization problem $\max\limits_P \operatorname{tr}(P^\top T_{ij})$.

Therefore, if we assume that the number of element sets n is not too large as there exists some constant $\gamma > 0$ such that $n \leq m^\gamma$, then by union bound we know that with probability at least $1 - m^{-\delta}$ for any $\delta > 0$ that we can guarantee that using the Pairwise Alignment method recovers a correct solution if $m \geq 2^{\frac{2}{4\gamma+\delta}}$ and if we set each $\eta_{ij} \leq \min(\frac{1}{10}, \frac{1}{2(2+4\gamma+\delta)\ln m+2}) = O(\frac{1}{\log m})$.

Next let us consider the probability that our Algorithm 1 recovers the correct permutation matrices. We would only make some errors on the updates on line 5 and 11. Basically, if we don't make any error at any iteration at the step of computing $\hat{P}$ on line 4 and don't make any updates on line 11, then we are sure that our algorithm solves the problem optimally.

Here we consider $m \geq 8$ such that $\frac{1}{10} > \frac{1}{4(1+\varepsilon)\ln m+2}$ for any $\varepsilon > 0$. Let us first bound the probability that we might make a mistake when computing the matrix $\hat{P}$ on line 4. At each iteration when we are considering edge (v_i, v_j), if we have $\eta_{ij} \leq \frac{1}{4(1+\varepsilon)\ln m+2}$ for any $\varepsilon > 0$, then from the above analysis we know that with probability at least $1 - m^{-2\varepsilon}$ we do not make mistakes on this step.

Otherwise, let us take $(i^*, j^*) = \operatorname*{argmin}\limits_{i' \in S_i, j' \in S_j} \eta_{i'j'}$ (S_i and S_j are the sets before being updated on line 7). If we have $\eta_{i^*j^*} \leq \frac{1}{8(1+\varepsilon)\ln m+4} \leq \frac{1}{4(1+\varepsilon)\ln m+2}$, then from the above analysis we know that with probability at least $1 - m^{-2\varepsilon}$ we get $\hat{T}_{i^*j^*}$ from the optimization problem $\max\limits_P \operatorname{tr}(P^\top T_{i^*j^*})$, and we also know that $\eta_{ij} - \eta_{i^*j^*} \geq \frac{1}{8(1+\varepsilon)\ln m+4}$.

Notice that (v_i, v_j) is an edge of the Maximum Spanning Tree of G. It must be the edge with largest edge weight between vertices in S_i and S_j. Therefore. we have $f(T_{ij}) \geq f(T_{i^*j^*})$. Conditioned on the cases that we recover $\hat{T}_{i^*j^*}$ from $\max\limits_P \operatorname{tr}(P^\top T_{i^*j^*})$ (we will omit some conditional probability notation from now on for brevity), and let $U \sim \chi^2(m)$ be a Chi-Square random variable with free degree m, then by the Chi-Square tail bounds that Laurent and Massart (2000) proposed,

$$\begin{aligned} &\Pr[f(T_{i^*j^*}) \leq m(1 - \eta_{i^*j^*} - \frac{1}{16(1+\varepsilon)\ln m + 8})] \\ &= \Pr[U - m \geq \frac{m}{\eta_{i^*j^*}(16(1+\varepsilon)\ln m + 8)}] \\ &\leq \Pr[U - m \geq \frac{m}{2}] \leq \Pr[U - m \geq 0.48m] \\ &\leq \exp(-\frac{m}{25}) \leq m^{-2\varepsilon} \end{aligned} \tag{11}$$

for sufficiently large m. On the other hand, consider the value of $f(T_{ij})$, denote $P' = \operatorname*{argmax}\limits_P \operatorname{tr}(P^\top T_{ij})$ and k to be the number of entries where $\hat{T}_{ij}$ equals 1 while P' does not equal 1 ($0 \leq k \leq m$). Since we require all $\eta_{ij} \leq O(1)$, let us assume that we have $\eta_{ij} \leq \frac{1}{3}$. Let $V_1 \sim \chi^2(k)$, $V_2 \sim \chi^2(m-k)$ be two independent Chi-Square random variables (we use $\chi^2(0)$ to be the random variable that only has support on a single point 0). Conditioned on k, the distribution of $f(T_{ij})$ is the same with $\eta_{ij}(V_1 - V_2) + m - k$. If $k > 0$, we know that

$$\begin{aligned} &\Pr[V_1 \geq k + \frac{m}{\eta_{ij}(32(1+\varepsilon)\ln m + 16)}] \\ \leq &\Pr[V_1 - k \geq \frac{3m}{32(1+\varepsilon)\ln m + 16}] \\ \leq &\Pr[V_1 - k \geq 2\sqrt{2k\varepsilon \ln m} + 4\varepsilon \ln m] \leq m^{-2\epsilon} \end{aligned} \tag{12}$$

for sufficiently large m. Symmetrically, if $k < m$,

$$\begin{aligned} &\Pr[V_2 \leq (m-k) - \frac{n}{\eta_{ij}(32(1+\varepsilon)\ln m + 16)}] \\ \leq &\Pr[(m-k) - V_2 \geq \frac{3m}{32(1+\varepsilon)\ln m + 16}] \\ \leq &\Pr[(m-k) - V_2 \geq 2\sqrt{2k\varepsilon \ln m}] \leq m^{-2\epsilon}. \end{aligned} \tag{13}$$

for sufficiently large m. Therefore, conditioned on k, if we have $\eta_{ij} \leq \frac{1}{3}$, we always have

$$\begin{aligned} &\Pr[\eta_{ij}(V_1 - V_2) + m - k \geq m(1 - \eta_{ij} + \frac{1}{16(1+\varepsilon)\ln m + 8})] \\ &\leq \Pr[\eta_{ij}(V_1 - V_2) - (2k - m)\eta_{ij} \geq \frac{1}{16(1+\varepsilon)\ln m + 8})] \\ &\leq \Pr[V_1 \geq k + \frac{m}{\eta_{ij}(32(1+\varepsilon)\ln m + 16)}] \\ &+ \Pr[V_2 \leq (m-k) - \frac{m}{\eta_{ij}(32(1+\varepsilon)\ln m + 16)}] \leq 2m^{-2\varepsilon}. \end{aligned} \tag{14}$$

This is true for all k. Hence, without conditioning on k, we know that

$$\Pr[f(T_{ij}) \geq m(1 - \eta_{ij} + \frac{1}{16(1+\varepsilon)\ln m + 8})] \leq 2m^{-2\varepsilon} \tag{15}$$

for sufficiently large m and if we have $\eta_{ij} \leq \frac{1}{3}$.

By union bound on Equations (11) and (15), we know that, conditioned on the cases where we recover $\hat{T}_{i^*j^*}$ from $\max_P \operatorname{tr}(P^\top T_{i^*j^*})$, since $\eta_{ij} - \eta_{i^*j^*} \geq \frac{1}{8(1+\varepsilon)\ln m+4}$, we have

$$\begin{aligned}&\Pr[f(T_{ij}) \geq f(T_{i^*j^*})]\\ \leq&\Pr[f(T_{i^*j^*}) \leq m(1-\eta_{i^*j^*} - \frac{1}{16(1+\varepsilon)\ln m+8})]\\ &+\Pr[f(T_{ij}) \geq m(1-\eta_{ij} + \frac{1}{16(1+\varepsilon)\ln m+8})]\\ \leq& 3m^{-2\varepsilon}.\end{aligned} \tag{16}$$

Since we know that, if we have $\eta_{i^*j^*} \leq \frac{1}{8(1+\varepsilon)\ln m+4}$, then with probability at least $1 - m^{-2\varepsilon}$ we would recover $\hat{T}_{i^*j^*}$ from $\max_P \operatorname{tr}(P^\top T_{i^*j^*})$. Hence, conditioned on the case that $\eta_{ij} > \frac{1}{4(1+\varepsilon)\ln m+2}$, we know that the probability $\Pr[f(T_{ij}) \geq f(T_{i^*j^*})] \leq 3m^{-2\varepsilon} + m^{-2\varepsilon} = 4m^{-2\varepsilon}$. Plus the opposite case where $\eta_{ij} \leq \frac{1}{4(1+\varepsilon)\ln m+2}$, by union bound we know that the probability that we make an error during each iteration of the initialization part of Algorithm 1 is at most $5m^{-2\varepsilon}$. This is true under the condition that $\eta_{ij} \leq \frac{1}{3}$ and $\min_{i' \in S_i, j' \in S_j} \eta_{ij} \leq \frac{1}{8(1+\varepsilon)\ln m+4}$. To make these two conditions true, we impose the following two requirements:

- Consider an undirected weighted graph $G' = (V', E'')$, where there is a vertex v_i' for each element set X_i and their is en edge $(v_i', v_j') \in E''$ with edge weight η_{ij}. Then the bottleneck weight of the minimum bottleneck spanning tree of G' should be at most $\frac{1}{8(1+\varepsilon)\ln m+4}$.
- $\max_{i<j} \eta_{ij} \leq \frac{1}{3}$.

Therefore, under the above two conditions, assume the number of element sets m satisfy $n \leq m^\gamma$ for some constant $\gamma > 0$. Then, by union bound we know that, for sufficiently large n, the probability that we recover the correct solution for $(\hat{A}_1, \ldots, \hat{A}_n)$ during the initialization part of the Algorithm 1 is $1 - 5m^{-2\varepsilon+\gamma}$.

For the coordinate update part of Algorithm 1 (line 9 to 12), let us consider the probability that we do not perform any updates conditioned on the case that we already have an optimal solution in the initialization part. For each step, denote the matrix we are optimizing as A_i. The update rule is Equation (3). Using the same approach as before, assume that there is some matrix $P' \neq A_i$ such that $\operatorname{tr}(P'^\top \sum_{1\leq j\leq n, i\neq j} A_j T_{ij}) \geq \operatorname{tr}(A_i^\top \sum_{1\leq j\leq n, i\neq j} A_j T_{ij})$. Denote k as the number of entries where A_i equals 1 but P' does not. Also, denote $U_1, \ldots, U_{i-1}, U_{i+1}, \ldots, U_n$ to be independent random variables following the distribution $\chi^2(2k)$, and let U be a random variable following distribution $\chi^2(2k(n-1))$. Then, by the Chi-Square tail bounds that Laurent and Massart (2000) proposed

$$\begin{aligned}&\Pr[\operatorname{tr}(P'^\top \sum_{1\leq j\leq n, i\neq j} A_j T_{ij}) \geq \operatorname{tr}(A_i^\top \sum_{1\leq j\leq n, i\neq j} A_j T_{ij})]\\ =&\Pr[\sum_{1\leq j\leq n, i\neq j} \eta_{ij} U_j \geq k(n-1)]\\ \leq&\Pr[\frac{1}{3}U \geq k(n-1)] \leq \exp(-\frac{2k(n-1)}{25}).\end{aligned} \tag{17}$$

Then, again by union bound on all values for k, we know that the probability that we might get a wrong answer for A_i in a single step is at most

$$\begin{aligned}&\sum_{k=2}^{m} p_k \cdot \frac{m!}{(m-k)!} \cdot \exp(-\frac{2k(n-1)}{25})\\ \leq&\frac{1}{2}\sum_{k=2}^{m} m^k \cdot \exp(-\frac{2k(n-1)}{25}).\end{aligned} \tag{18}$$

If we have $n \geq 20\ln m$, then for sufficient large value of m we have

$$\frac{1}{2}\sum_{k=2}^{m} m^k \cdot \exp(-\frac{2k(n-1)}{25}) \leq m^{-2}. \tag{19}$$

By union bound on all m matrices A_i's, we know that the probability at least one of them needs updates is at most m^{-1}. Hence, we can solve the optimization problem with probability at least $1 - 5m^{-2\varepsilon+\gamma} + m^{-1}$ under all of the above constraints. If we set $\varepsilon = \frac{\gamma+1}{2}$, then the probability becomes $1 - 6m^{-1} = 1 - o(1)$ for sufficiently large m. Under that setting, we require the bottleneck weight of the minimum bottleneck spanning tree of G' to be at at most $\frac{1}{8(1+\varepsilon)\ln m+4} = \frac{1}{4(3+\gamma)\ln m+4}$. □