A PROOFS

A.1 Proof of Theorem 1

The problem is clearly in NP, since the cost of the solution could be checked according to the definition of the column-groups storage in definition 4 in polynomial time.

To show the NP-completeness, we will build a reduction from the k-set packing problem, one of the Karp's 21 NP-complete problems [15].

Given a finite set $\mathbf{E}$, and a list $\mathcal{L}$ of the subsets of $\mathbf{E}$, i.e., $\forall \mathbf{L} \in \mathcal{L}, \mathbf{L} \subseteq \mathbf{E}$. The set packing problem is to determine whether there exist some k or more subsets in $\mathcal{L}$ are pairwise disjoint. Next, we will start from an instance of the set packing problem, to our column-grouping problem. Combining with that the NP is in NP, we can conclude the problem is NP-complete.

Given an instance of the k-set packing problem, i.e., given $\mathbf{E} = \{e_1, e_2, \ldots, e_m\}$ that are the elements of $\mathbf{E}$, and given a list $\mathcal{L}$ of the subsets of $\mathbf{E}$. We first let the time series set $\mathbf{S}$ in our column-groups storage problem corresponds to $\mathbf{E}$, i.e., S_i corresponds to e_i, $1 \le i \le m$. Next, we transform our problem from computing $cost_{cg}(\mathcal{G}, \mathbf{R})$ into computing $cost_c(\mathbf{S}) - cost_{cg}(\mathcal{G}, \mathbf{R})$, which is equivalent, since $cost_c(\mathbf{S})$ is a constant determined by the input $\mathbf{S}$. Hence, since

$$cost_c(\mathbf{S}) - cost_{cg}(\mathcal{G}, \mathbf{R}) = \sum_{\mathbf{G} \in \mathcal{G}} \big(cost_c(\mathbf{G}) - cost_g(\mathbf{G})\big)$$

We thus transform our problem into computing the difference of using single group or using single column for each group $\mathbf{G}$. Next, we let $\mathcal{G}$ correspond to $\mathcal{L}$, i.e., for $\mathbf{L} \in \mathcal{L}$, by assigning timestamps for $\mathbf{G}$ so that $cost_c(\mathbf{G}) - cost_g(\mathbf{G}) = 1$ (this could be created by only allowing only one timestamp is aligned, and β is set to 0 for simplicity). That is to say, a solution $\mathcal{G}^*$ for our problem with $\sum_{\mathbf{G} \in \mathcal{G}} \big(cost_c(\mathbf{G}) - cost_g(\mathbf{G})\big) \ge k$ is exactly the solution for k-set packing problem of size k or more. Therefore, an instance of the k-set packing problem is exactly an instance of our problem.

In summary, combining with the problem could be verified in polynomial time, we prove the NP-completeness of our problem.

A.2 Proof of Lemma 2

Proof sketch. Let S_1, S_2 be two columns to be merged, we have following formula:

$$\begin{aligned} \Delta(S_1, S_2) &= (m_{S_1} + m_{S_2} - m_{\{S_1, S_2\}})\alpha - 2m_{\{S_1, S_2\}}\beta \\ &= P(S_1, S_2)\alpha - 2(m_{S_1} + m_{S_2} - P(S_1, S_2))\beta \end{aligned} \tag{18}$$

According to the principle of $\Delta(S_1, S_2) > 0$ we have

$$P(S_1, S_2)\alpha - 2(m_{S_1} + m_{S_2} - P(S_1, S_2))\beta > 0 \tag{19}$$

By rearranging the formula, we obtain the constraint for $P(S_1, S_2)$:

$$P(S_1, S_2) > \frac{2(m_{S_1} + m_{S_2})\beta}{\alpha + 2\beta} \tag{20}$$

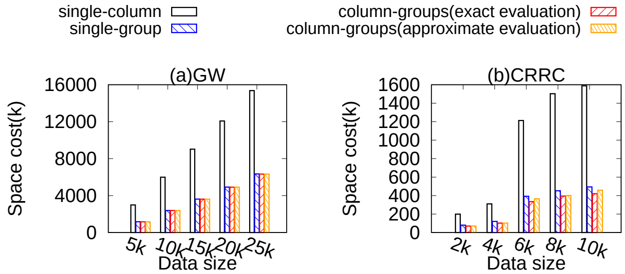

Figure 23: Space cost by varying data size

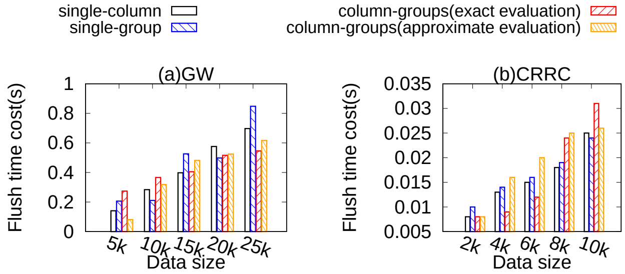

Figure 24: Flush time by varying data size

A.3 Proof of Lemma 3

Proof sketch. For column-group merging, let $S_1, \mathbf{G}_1$ be the column and the group to be merged, we have following formula:

$$\begin{aligned} \Delta(S_1, \mathbf{G}_1) &= (m_{S_1} + m_{\mathbf{G}_1} - m_{\{S_1\} \cup \mathbf{G}_1})\alpha \\ &+ (n_{\mathbf{G}_1} m_{\mathbf{G}_1} - (n_{\mathbf{G}_1} + 1) m_{\{S_1\} \cup \mathbf{G}_1})\beta \\ &= P(S_1, \mathbf{G}_1)\alpha + (n_{\mathbf{G}_1} m_{\mathbf{G}_1} - (n_{\mathbf{G}_1} + 1)(m_{S_1} \\ &+ m_{\mathbf{G}_1} - P(S_1, \mathbf{G}_1))\beta \end{aligned} \tag{21}$$

Similarly, by combining with $\Delta(S_1, \mathbf{G}_1) > 0$ and rearranging the formula, we obtain the constraint for $P(S_1, \mathbf{G}_1)$:

$$\begin{aligned} P(S_1, \mathbf{G}_1) &> \frac{((n_{\mathbf{G}_1} + 1)(m_{S_1} + m_{\mathbf{G}_1}) - n_{\mathbf{G}_1} m_{\mathbf{G}_1})\beta}{\alpha + (n_{\mathbf{G}_1} + 1)\beta} \\ &= \frac{(n_{\mathbf{G}_1} m_{S_1} + m_{\mathbf{G}_1} + m_{S_1})\beta}{\alpha + (n_{\mathbf{G}_1} + 1)\beta} \end{aligned} \tag{22}$$

A.4 Proof of Lemma 4

Proof sketch. For group-group merging, let $\mathbf{G}_1, \mathbf{G}_2$ be the groups to be merged, the merging gain should be

$$\begin{aligned} \Delta(\mathbf{G}_1, \mathbf{G}_2) &= (m_{\mathbf{G}_1} + m_{\mathbf{G}_2} - m_{\mathbf{G}_1 \cup \mathbf{G}_2})\alpha \\ &+ (n_{\mathbf{G}_1} m_{\mathbf{G}_1} + n_{\mathbf{G}_2} m_{\mathbf{G}_2} - (n_{\mathbf{G}_1} + n_{\mathbf{G}_2}) m_{\mathbf{G}_1 \cup \mathbf{G}_2})\beta \\ &= P(\mathbf{G}_1, \mathbf{G}_2)\alpha + ((n_{\mathbf{G}_1} + n_{\mathbf{G}_2}) P(\mathbf{G}_1, \mathbf{G}_2) + n_{\mathbf{G}_1} m_{\mathbf{G}_2} + n_{\mathbf{G}_2} m_{\mathbf{G}_1})\beta \end{aligned} \tag{23}$$

Similarly, by combining with $\Delta(\mathbf{G}_1, \mathbf{G}_2) > 0$ and rearranging the formula, we obtain the constraint for $P(\mathbf{G}_1, \mathbf{G}_2)$:

$$P(\mathbf{G}_1, \mathbf{G}_2) > \frac{(n_{\mathbf{G}_1} m_{\mathbf{G}_2} + n_{\mathbf{G}_2} m_{\mathbf{G}_1})\beta}{\alpha + (n_{\mathbf{G}_1} + n_{\mathbf{G}_2})\beta} \tag{24}$$

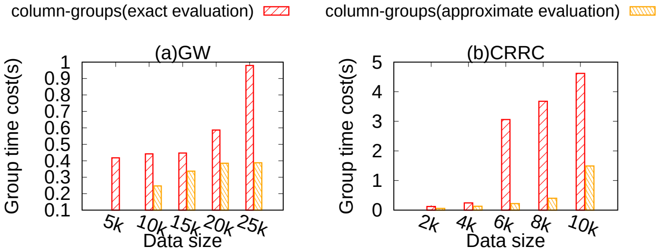

Figure 25: Group time by varying data size

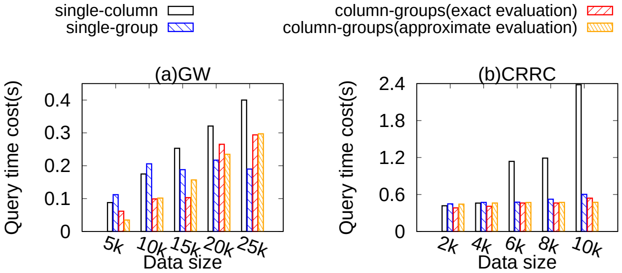

Figure 26: Query time cost by varying data size

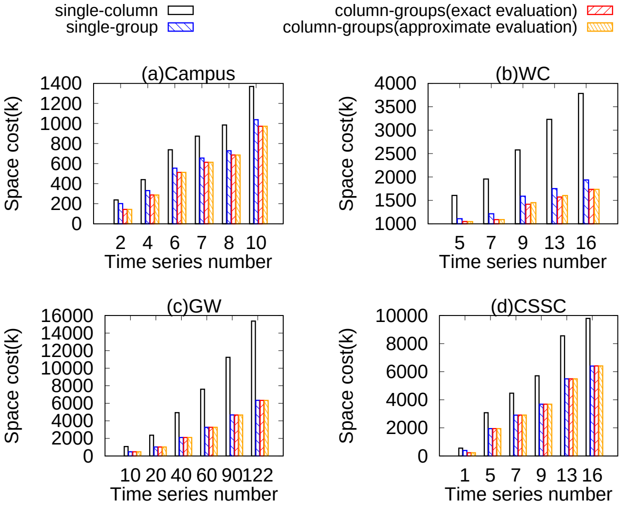

Figure 27: Space cost by varying time series number

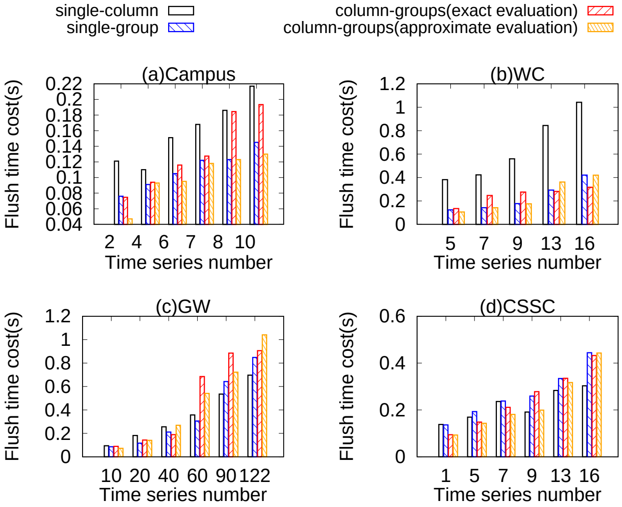

Figure 28: Flush time cost by varying time series number

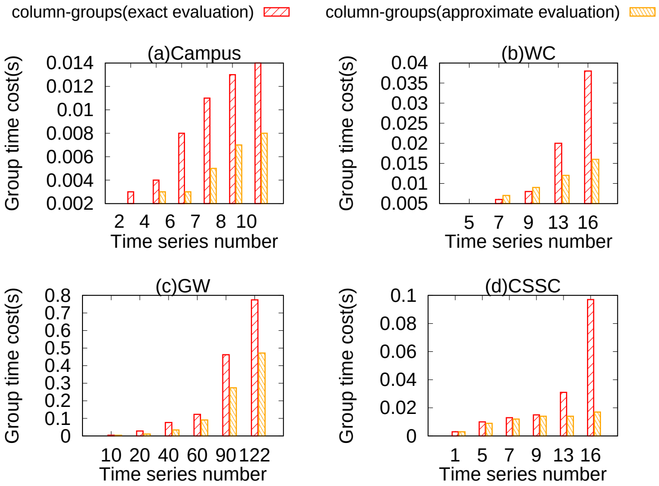

Figure 29: Group time cost by varying time series number

B ADDITIONAL RESULTS

B.1 Evaluation of Different Alignment Degree

As introduced in Section 2, the grouping problem is between the space cost for bitmaps and timestamps. The intuition is that, if the timestamps of different time series are highly aligned, which could generate the largest merging gain, they tend to be grouped. To this end, we choose a highly aligned dataset GW, and introduce noises of various rates into each time series, denoted by noise rate, from 1% to 25%. For instance, noise rate 1% denotes that for each time series, we randomly choose 1% data points and modify their timestamps with random noises. That is, we manually make the dataset unaligned with different degrees by introducing the noises.

The results are shown in Figure 31. It is not surprising that, the performance of the single-group storage is highly sensitive to the increasing noise rate, since the noises make it difficult for alignment, which significantly increases the space cost for storing the bitmaps. In contrast, the proposed column-groups storage benefits from the grouping strategy which mainly merges time series with positive merging gains, and thus shows robustness over different noise rates. In addition, column-groups with exact evaluation shows better robustness to noise. Since the single-column storage stores time series in column order separately, the space cost is not significantly affected by the noise rate.

Figures 31(b), 31(c), 31(d) show the time cost for flushing, grouping and querying, respectively. The single-group scheme shows

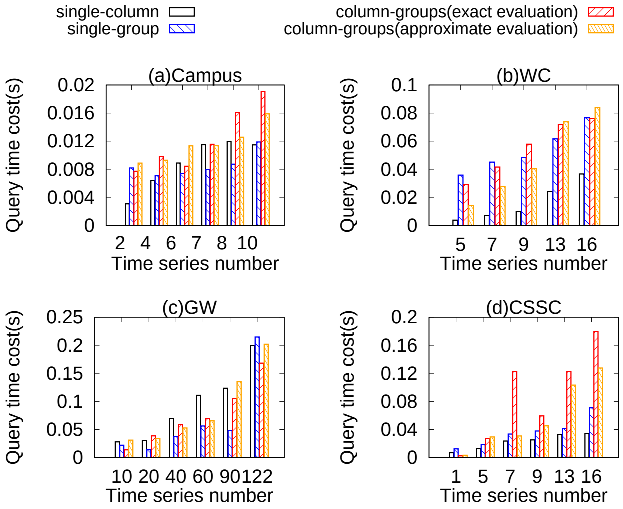

Figure 30: Query time cost by varying time series number

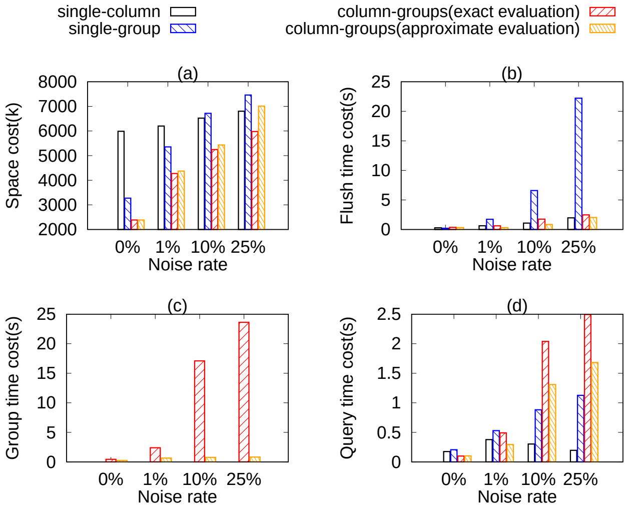

Figure 31: Varying noise rate over GW dataset

larger flush time, since it is expensive to align all the noisy timestamps when flushing. The column-groups storage scheme with exact evaluation shows significantly larger group time cost, since the noise increases the hardness to compute overlaps, while the approximate evaluation is more efficient. The query time of all the grouping schemes is affected by high noise rate.