Appendix

A. Proofs

Proof of Lemma 1:

It is straightforward to see that Algorithm 1 can be implemented in time $\mathcal{O}((k+|C_0'|)|S|)$. We only need to show that it is a 2-approximation algorithm for (3).

If $k=0$, there is nothing to show, so assume that $k\geq 1$. Let $C=\{c_1,\ldots,c_k\}$ be the output of Algorithm 1 and $C^*=\{c_1^*,\ldots,c_k^*\}$ be an optimal solution to (3) with objective value r^*. Let $s\in S$ be arbitrary. We need to show that $d(s,\hat{c})\leq 2r^*$ for some $\hat{c}\in C\cup C_0'$. If $s\in C\cup C_0'$, there is nothing to show. So assume $s\notin C\cup C_0'$. If

$$C_0'\cap \operatorname*{argmin}_{c\in C^*\cup C_0'} d(s,c)\neq\emptyset,$$

there exists $\hat{c}\in C_0'$ with $d(s,\hat{c})\leq r^*$ and we are done. Otherwise, let $c_i^*\in\operatorname{argmin}_{c\in C^*\cup C_0'} d(s,c)$ and hence $d(s,c_i^*)\leq r^*$. We distinguish two cases:

- $\exists\, c_j\in C$ with $c_i^*\in\operatorname{argmin}_{c\in C^*\cup C_0'} d(c_j,c)$:

  We have $d(c_j,c_i^*)\leq r^*$ and hence $d(s,c_j)\leq d(s,c_i^*)+d(c_i^*,c_j)\leq 2r^*$.

- $\nexists\, c_j\in C$ with $c_i^*\in\operatorname{argmin}_{c\in C^*\cup C_0'} d(c_j,c)$:

  There must be $c'\neq c''\in C\cup C_0'$, where not both c' and c'' can be in C_0', and $\hat{c}\in C^*\cup C_0'$ such that

  $$\hat{c}\in \operatorname*{argmin}_{c\in C^*\cup C_0'} d(c',c)\cap \operatorname*{argmin}_{c\in C^*\cup C_0'} d(c'',c).$$

  Since $d(c',\hat{c})\leq r^*$ and $(c'',c^*)\leq r^*$, it follows that $d(c',c'')\leq d(c',\hat{c})+d(\hat{c},c'')\leq 2r^*$.

  Without loss of generality, assume that in the execution of Algorithm 1, c'' has been added to the set of centers after c' has been added. In particular, we have $c''\in C$ and $c''=c_l$ for some $l\in\{1,\ldots,k\}$. Due to the greedy choice in Line 5 of the algorithm and since s has not been chosen by the algorithm, we have

  $$2r^*\geq d(c',c'')\geq \min_{c\in\{c_1,\ldots,c_{l-1}\}\cup C_0'} d(c'',c)\geq \min_{c\in\{c_1,\ldots,c_{l-1}\}\cup C_0'} d(s,c).$$

□

Proof of Theorem 1:

Again it is easy to see that Algorithm 2 can be implemented in time $\mathcal{O}((k+|C_0|)|S|)$. We need to prove that it is a 5-approximation algorithm, but not a $(5-\varepsilon)$-approximation algorithm for any $\varepsilon>0$:

1. Algorithm 2 is a 5-approximation algorithm:

   Let r^*_{fair} be the optimal value of the fair problem (2) and r^* be the optimal value of the unfair problem (3). Clearly, $r^*\leq r^*_{\text{fair}}$. Let $C^*_{\text{fair}}=\{c_1^{(1)*},\ldots,c_{k_{S_1}}^{(1)*},c_1^{(2)*},\ldots,c_{k_{S_2}}^{(2)*}\}$ with $c_1^{(1)*},\ldots,c_{k_{S_1}}^{(1)*}\in S_1$ and $c_1^{(2)*},\ldots,c_{k_{S_2}}^{(2)*}\in S_2$ be an optimal solution to the fair problem (2) with cost r^*_{fair} and $C^A=\{c_1^A,\ldots,c_k^A\}$ be the centers returned by Algorithm 2. It is clear that Algorithm 2 returns k_{S_1} many elements from S_1 and k_{S_2} many elements from S_2 and hence $C^A=\{c_1^{(1)A},\ldots,c_{k_{S_1}}^{(1)A},c_1^{(2)A},\ldots,c_{k_{S_2}}^{(2)A}\}$ with $c_1^{(1)A},\ldots,c_{k_{S_1}}^{(1)A}\in S_1$ and $c_1^{(2)A},\ldots,c_{k_{S_2}}^{(2)A}\in S_2$. We need to show that

   $$\min_{c\in C^A\cup C_0} d(s,c)\leq 5r^*_{\text{fair}},\quad s\in S.$$

Let $\widetilde{C}^A = \{\tilde{c}_1^A, \ldots, \tilde{c}_k^A\}$ be the output of Algorithm 1 when called in Line 3 of Algorithm 2. Since Algorithm 1 is a 2-approximation algorithm for the unfair problem (3) according to Lemma 1, we have

$$\min_{c \in \widetilde{C}^A \cup C_0} d(s,c) \leq 2r^* \leq 2r^*_{\text{fair}}, \quad s \in S. \tag{6}$$

If Algorithm 2 returns $\widetilde{C}^A$ in Line 6, that is $C^A = \widetilde{C}^A$, we are done. Otherwise assume, as in the algorithm, that $|\widetilde{C}^A \cap S_1| > k_{S_1}$. Let $\tilde{c}_i^A \in S_1$ be a center of cluster L_i that we replace with $y \in L_i \cap S_2$ and let $\hat{y}$ be an arbitrary element in L_i. Because of (6), we have $d(\tilde{c}_i^A, y) \leq 2r^*_{\text{fair}}$ and $d(\tilde{c}_i^A, \hat{y}) \leq 2r^*_{\text{fair}}$, and hence $d(y, \hat{y}) \leq d(y, \tilde{c}_i^A) + d(\tilde{c}_i^A, \hat{y}) \leq 4r^*_{\text{fair}}$ due to the triangle inequality. Consequently, after the while-loop in Line 9, every $s \in S$ is in distance of $4r^*_{\text{fair}}$ or smaller to the center of its cluster. In particular, we have

$$\min_{c \in \widetilde{C}^A \cup C_0} d(s,c) \leq 4r^*_{\text{fair}}, \quad s \in S,$$

and if Algorithm 2 returns $\widetilde{C}^A$ in Line 13, we are done. Otherwise, we still have $|\widetilde{C}^A \cap S_1| > k_{S_1}$ after exchanging centers in the while-loop in Line 9. Let $S' = \cup_{i \in [k]: \tilde{c}_i^A \in S_1} L_i$, that is the union of clusters with a center $\tilde{c}_i^A \in S_1$. Since there is no more center in S_1 that we can exchange for an element in S_2, we have $S' \subseteq S_1$. Let $S'' = \cup_{i \in [k]: \tilde{c}_i^A \in S_2} L_i$ be the union of clusters with a center $\tilde{c}_i^A \in S_2$ and $S_{C_0} = L'_1 \cup \ldots \cup L'_{|C_0|}$ be the union of clusters with a center in C_0. Then we have $S = S' \,\dot{\cup}\, S'' \,\dot{\cup}\, S_{C_0}$. We have $\widetilde{C}^A \cap S_2 \subseteq C^A$ and

$$\min_{c \in C^A \cup C_0} d(s,c) \leq \min_{c \in (\widetilde{C}^A \cap S_2) \cup C_0} d(s,c) \leq 4r^*_{\text{fair}}, \quad s \in S'' \cup S_{C_0}. \tag{7}$$

Hence we only need to show that $\min_{c \in C^A \cup C_0} d(s,c) \leq 5r^*_{\text{fair}}$ for every $s \in S'$. We split S' into two subsets $S' = S'_a \dot{\cup} S'_b$, where

$$S'_a = \left\{ s \in S' : \operatorname*{argmin}_{c \in C^*_{\text{fair}} \cup C_0} d(s,c) \,\cap\, (C_0 \cup S_2) \neq \emptyset \right\}$$

and $S'_b = S' \setminus S'_a$. For every $s \in S'_a$ there is $c \in (C_0 \cup S_2) \subseteq (S'' \cup S_{C_0})$ with $d(s,c) \leq r^*_{\text{fair}}$ and it follows from (7) and the triangle inequality that

$$\min_{c \in C^A \cup C_0} d(s,c) \leq \min_{c \in (\widetilde{C}^A \cap S_2) \cup C_0} d(s,c) \leq 5r^*_{\text{fair}}, \quad s \in S'_a. \tag{8}$$

It remains to show that $\min_{c \in C^A \cup C_0} d(s,c) \leq 5r^*_{\text{fair}}$ for every $s \in S'_b$. For every $s \in S'_b$ there exists $c \in \{c_1^{(1)*}, \ldots, c_{k_{S_1}}^{(1)*}\}$ with $d(s,c) \leq r^*_{\text{fair}}$. We can write $S'_b = \cup_{j=1}^{k_{S_1}} \{s \in S'_b : d(s, c_j^{(1)*}) \leq r^*_{\text{fair}}\}$ (some of the sets in this union might be empty, but that does not matter). Note that for every $j \in \{1, \ldots, k_{S_1}\}$ we have

$$d(s,s') \leq 2r^*_{\text{fair}}, \quad s, s' \in \left\{ s \in S'_b : d(s, c_j^{(1)*}) \leq r^*_{\text{fair}} \right\}, \tag{9}$$

due to the triangle inequality. It is

$$S' = S'_a \cup S'_b = S'_a \cup \bigcup_{j=1}^{k_{S_1}} \left\{ s \in S'_b : d(s, c_j^{(1)*}) \leq r^*_{\text{fair}} \right\}$$

and when, in Line 15 of Algorithm 2, we run Algorithm 1 on $S' \cup C'_0$ with $k = k_{S_1}$ and initial centers $C'_0 = C_0 \cup (\widetilde{C}^A \cap S_2)$, one of the following three cases has to happen (we denote the centers returned by Algorithm 1 by $\widehat{C}^A = \{c_1^{(1)A}, \ldots, c_{k_{S_1}}^{(1)A}\}$):

- For every $j \in \{1, \ldots, k_{S_1}\}$ there exists $j' \in \{1, \ldots, k_{S_1}\}$ such that $c_{j'}^{(1)A} \in \{s \in S'_b : d(s, c_j^{(1)*}) \leq r^*_{\text{fair}}\}$. In this case it immediately follows from (9) that

$$\min_{c \in C^A \cup C_0} d(s,c) \leq \min_{c \in \widehat{C}^A} d(s,c) \leq 2r^*_{\text{fair}}, \quad s \in S'_b.$$

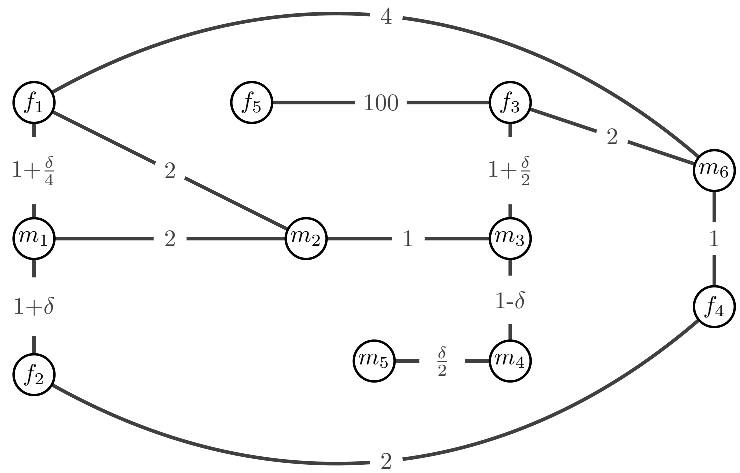

Figure 7. An example showing that Algorithm 2 is not a $(5-\varepsilon)$-approximation algorithm for any $\varepsilon > 0$.

- There exists $j' \in \{1, \ldots, k_{S_1}\}$ such that $c_{j'}^{(1)A} \in S'_a$. When Algorithm 1 picks $c_{j'}^{(1)A}$, any other element in S' cannot be at a larger minimum distance from a center in $(\widetilde{C}^A \cap S_2) \cup C_0$ or a previously chosen center in $\widehat{C}^A$ than $c_{j'}^{(1)A}$. It follows from (8) that

$$\min_{c \in C^A \cup C_0} d(s,c) \leq 5r^*_{\text{fair}}, \quad s \in S'.$$

- There exist $j \in \{1, \ldots, k_{S_1}\}$ and $j' \neq j'' \in \{1, \ldots, k_{S_1}\}$ such that $c_{j'}^{(1)A}, c_{j''}^{(1)A} \in \{s \in S'_b : d(s, c_j^{(1)*}) \leq r^*_{\text{fair}}\}$. Assume that Algorithm 1 picks $c_{j'}^{(1)A}$ before $c_{j''}^{(1)A}$. When Algorithm 1 picks $c_{j''}^{(1)A}$, any other element in S' cannot be at a larger minimum distance from a center in $(\widetilde{C}^A \cap S_2) \cup C_0$ or a previously chosen center in $\widehat{C}^A$ than $c_{j''}^{(1)A}$. Because of $d(c_{j'}^{(1)A}, c_{j''}^{(1)A}) \leq 2r^*_{\text{fair}}$ according to (9), it follows that

$$\min_{c \in C^A \cup C_0} d(s,c) \leq 2r^*_{\text{fair}}, \quad s \in S'.$$

In all cases we have

$$\min_{c \in C^A \cup C_0} d(s,c) \leq 5r^*_{\text{fair}}, \quad s \in S'_b,$$

which completes the proof of the claim that Algorithm 2 is a 5-approximation algorithm.

2. Algorithm 2 is not a $(5-\varepsilon)$-approximation algorithm for any $\varepsilon > 0$:

Consider the example given by the weighted graph shown in Figure 7, where $0 < \delta < \frac{1}{10}$. We have $S = S_1 \dot{\cup} S_2$ with $S_1 = \{f_1, f_2, f_3, f_4, f_5\}$ and $S_2 = \{m_1, m_2, m_3, m_4, m_5, m_6\}$. All distances are shortest-path-distances. Let $k_{S_1} = 1$, $k_{S_2} = 3$, and $C_0 = \emptyset$. We assume that Algorithm 1 in Line 3 of Algorithm 2 picks f_5 as first center. It then chooses f_2 as second center, f_3 as third center and f_1 as fourth center. Hence, $\widetilde{C}^A = \{f_5, f_2, f_3, f_1\}$ and $|\widetilde{C}^A \cap S_1| > k_{S_1}$. The clusters corresponding to $\widetilde{C}^A$ are $\{f_5\}$, $\{f_2, f_4\}$, $\{f_3, m_3, m_4, m_5, m_6\}$ and $\{f_1, m_1, m_2\}$. Assume we replace f_3 with m_4 and f_1 with m_2 in Line 10 of Algorithm 2. Then it is still $|\widetilde{C}^A \cap S_1| > k_{S_1}$, and in Line 15 of Algorithm 2 we run Algorithm 1 on $\{f_2, f_4, f_5\} \cup \{m_2, m_4\}$ with $k = 1$ and initially given centers $C'_0 = \{m_2, m_4\}$. Algorithm 1 returns $\widehat{C}^A = \{f_5\}$. Finally, assume that m_5 is chosen as arbitrary third center from S_2 in Line 16 of Algorithm 2. So the centers returned by Algorithm 2 are $C^A = \{f_5, m_2, m_4, m_5\}$ with a cost of $5 - \frac{\delta}{2}$ (incurred for f_4). However, the optimal solution $C^*_{\text{fair}} = \{f_5, m_1, m_3, m_6\}$ has cost only $1 + \delta$. Choosing δ sufficiently small shows that Algorithm 2 is not a $(5-\varepsilon)$-approximation algorithm for any $\varepsilon > 0$.

□

Proof of Lemma 2:

We want to show three things:

1. Algorithm 3 is well-defined:

   If the condition of the while-loop in Line 7 is true, there exists a shortest path $P = S_{v_0} S_{v_1} \cdots S_{v_w}$ with $S_{v_0} = S_r$, $S_{v_w} = S_s$ that connects S_r to S_s in G. Since P is a shortest path, all S_{v_i} are distinct. By the definition of G, for every $l = 0, \ldots, w-1$ there exists L_t with center $\tilde{c}_t^A \in S_{v_l}$ and $y \in L_t \cap S_{v_{l+1}}$. Hence, the for-loop in Line 8 is well defined.

2. Algorithm 3 terminates:

   Let, at the beginning of the execution of Algorithm 3 in Line 3, $H_1 = \{S_j \in \{S_1, \ldots, S_m\} : \tilde{k}_{S_j} = k_{S_j}\}$, $H_2 = \{S_j \in \{S_1, \ldots, S_m\} : \tilde{k}_{S_j} > k_{S_j}\}$ and $H_3 = \{S_j \in \{S_1, \ldots, S_m\} : \tilde{k}_{S_j} < k_{S_j}\}$. For $S_j \in H_1$, $\tilde{k}_{S_j}$ never changes during the execution of the algorithm. For $S_j \in H_2$, $\tilde{k}_{S_j}$ never increases during the execution of the algorithm and decreases at most until it equals k_{S_j}. For $S_j \in H_3$, $\tilde{k}_{S_j}$ never decreases during the execution of the algorithm and increases at most until it equals k_{S_j}. In every iteration of the while-loop, there is $S_j \in H_3$ for which $\tilde{k}_{S_j}$ increases by one. It follows that the number of iterations of the while-loop is upper-bounded by k.

3. Algorithm 3 exchanges centers in such a way that the set $\mathcal{G}$ that it returns satisfies $\mathcal{G} \subsetneq \{S_1, \ldots, S_m\}$ and properties (4) and (5):

   Note that throughout the execution of Algorithm 3 we have $\tilde{k}_{S_j} = \sum_{i=1}^{k} \mathbb{1}\left\{\tilde{c}_i^A \in S_j\right\}$ for the current centers $\tilde{c}_1^A, \ldots, \tilde{c}_k^A$. If the condition of the if-statement in Line 13 is true, then $\mathcal{G} = \emptyset$ and (4) and (5) are satisfied.

   Assume that the condition of the if-statement in Line 13 is not true. Clearly, the set $\mathcal{G}$ returned by Algorithm 3 satisfies (5). Since the condition of the if-statement in Line 13 is not true, there exist S_j with $\tilde{k}_{S_j} > k_{S_j}$ and S_i with $\tilde{k}_{S_i} < k_{S_i}$. We have $S_j \in \mathcal{G}$, but since the condition of the while-loop in Line 7 is not true, we cannot have $S_i \in \mathcal{G}$. This shows that $\mathcal{G} \subsetneq \{S_1, \ldots, S_m\}$. We need to show that (4) holds. Let L_h be a cluster with center $\tilde{c}_h^A \in S_f$ for some $S_f \in \mathcal{G}$ and assume it contained an element $o \in S_{f'}$ with $S_{f'} \notin \mathcal{G}$. But then we had a path from S_f to $S_{f'}$ in G. If $S_f \in \mathcal{G}'$, this is an immediate contradiction to $S_{f'} \notin \mathcal{G}$. If $S_f \notin \mathcal{G}'$, since $S_f \in \mathcal{G}$, there exists $S_g \in \mathcal{G}'$ such that there is a path from S_g to S_f. But then there is also a path from S_g to $S_{f'}$, which is a contradiction to $S_{f'} \notin \mathcal{G}$.

$\square$

Proof of Theorem 2:

For showing that Algorithm 4 is a $(3 \cdot 2^{m-1} - 1)$-approximation algorithm let r^*_{fair} be the optimal value of problem (2) and C^*_{fair} be an optimal solution with cost r^*_{fair}. Let C^A be the centers returned by Algorithm 4. A simple proof by induction over m shows that C^A actually comprises k_{S_i} many elements from every group S_i. We need to show that

$$\min_{c \in C^A \cup C_0} d(s,c) \leq (3 \cdot 2^{m-1} - 1) r^*_{\text{fair}}, \quad s \in S. \tag{10}$$

Let T be the total number of calls of Algorithm 4, that is we have one initial call and $T-1$ recursive calls. Since with each recursive call the number of groups is decreased by at least one, we have $T \leq m$. For $1 \leq j \leq T$, let $S^{(j)}$ be the data set in the j-th call of Algorithm 4. We additionally set $S^{(T+1)} = \emptyset$. We have $S^{(1)} = S$ and $S^{(j)} \supseteq S^{(j+1)}$, $1 \leq j \leq T$. For $1 \leq j < T$, let $\mathcal{G}^{(j)}$ be the set of groups in $\mathcal{G}$ returned by Algorithm 3 in Line 8 in the j-th call of Algorithm 4. If in the T-th call of Algorithm 4 the algorithm terminates from Line 10 (note that in this case we must have $T < m$), we also let $\mathcal{G}^{(T)} = \emptyset$ be the set of groups in $\mathcal{G}$ returned by Algorithm 3 in the T-th call. Otherwise we leave $\mathcal{G}^{(T)}$ undefined. Setting $\mathcal{G}^{(0)} = \{S_1, \ldots, S_m\}$, we have $\mathcal{G}^{(j)} \supsetneq \mathcal{G}^{(j+1)}$ for all j such that $\mathcal{G}^{(j+1)}$ is defined. For $1 \leq j < T$, let C_j be the set of centers returned by Algorithm 3 in Line 8 in the j-th call of Algorithm 4 that belong to a group not in $\mathcal{G}^{(j)}$ (in Algorithm 4, the set of these centers is denoted by C'). We analogously define C_T if in the T-th call of Algorithm 4 the algorithm terminates from Line 10. Note that the centers in C_j are comprised in the final output C^A of Algorithm 4, that is

$C_j \subseteq C^A$ for $1 \leq j < T$ or $1 \leq j \leq T$. As always, C_0 denotes the set of centers that are given initially (for the initial call of Algorithm 4). Note that in the j-th call of Algorithm 4 the set of initially given centers is $C_0 \cup \bigcup_{l=1}^{j-1} C_l$.

We first prove by induction that for all $j \geq 1$ such that $\mathcal{G}^{(j)}$ is defined, that is $1 \leq j < T$ or $1 \leq j \leq T$, we have

$$\min_{c \in C_0 \cup \bigcup_{l=1}^{j} C_l} d(s,c) \leq (2^{j+1} + 2^j - 2) r^*_{\text{fair}}, \quad s \in \left(S^{(j)} \setminus S^{(j+1)}\right) \cup \left(C_0 \cup \bigcup_{l=1}^{j} C_l\right). \tag{11}$$

Base case $j = 1$: In the first call of Algorithm 4, Algorithm 1, when called in Line 3 of Algorithm 4, returns an approximate solution to the unfair problem (3). Let $r^* \leq r^*_{\text{fair}}$ be the optimal cost of (3). Since Algorithm 1 is a 2-approximation algorithm for (3) according to Lemma 1, after Line 3 of Algorithm 4 we have

$$\min_{c \in \widetilde{C}^A \cup C_0} d(s,c) \leq 2r^* \leq 2r^*_{\text{fair}}, \quad s \in S.$$

Let $\tilde{c}_i^A \in \widetilde{C}^A$ be a center and $s_1, s_2 \in L_i$ be two points in its cluster. It follows from the triangle inequality that $d(s_1, s_2) \leq d(s_1, \tilde{c}_i^A) + d(\tilde{c}_i^A, s_2) \leq 4r^*_{\text{fair}}$. Hence, after running Algorithm 3 in Line 8 of Algorithm 4 and exchanging some of the centers in $\widetilde{C}^A$, we have $d(s, c(s)) \leq 4r^*_{\text{fair}}$ for every $s \in S$, where $c(s)$ denotes the center of its cluster. In particular,

$$\min_{c \in C_0 \cup C_1} d(s,c) \leq (2^{1+1} + 2^1 - 2) r^*_{\text{fair}} = 4r^*_{\text{fair}}$$

for all $s \in S$ for which its center $c(s)$ is in C_0 or in a group not in $\mathcal{G}^{(1)}$, that is for $s \in (S^{(1)} \setminus S^{(2)}) \cup (C_0 \cup C_1)$.

Inductive step $j \mapsto j+1$: Recall property (4) of a set $\mathcal{G}$ returned by Algorithm 3. Consequently, $S^{(j+1)}$ only comprises items in a group in $\mathcal{G}^{(j)}$ and, additionally, the given centers $C_0 \cup \bigcup_{l=1}^{j} C_l$.

We split $S^{(j+1)}$ into two subsets $S^{(j+1)} = S_a^{(j+1)} \dot\cup S_b^{(j+1)}$, where

$$S_a^{(j+1)} = \left\{ s \in S^{(j+1)} : \operatorname*{argmin}_{c \in C^*_{\text{fair}} \cup C_0} d(s,c) \;\cap\; \left(C_0 \cup \bigcup_{W \in \{S_1, \ldots, S_m\} \setminus \mathcal{G}^{(j)}} W \right) \neq \emptyset \right\}$$

and $S_b^{(j+1)} = S^{(j+1)} \setminus S_a^{(j+1)}$. For every $s \in S_a^{(j+1)}$ there exists

$$c \in C_0 \cup \bigcup_{W \in \{S_1, \ldots, S_m\} \setminus \mathcal{G}^{(j)}} W \subseteq \left(S \setminus S^{(j+1)}\right) \cup \left(C_0 \cup \bigcup_{l=1}^{j} C_l\right)$$

with $d(s,c) \leq r^*_{\text{fair}}$. It follows from the inductive hypothesis that there exists $c' \in C_0 \cup \bigcup_{l=1}^{j} C_l$ with $d(c,c') \leq (2^{j+1} + 2^j - 2) r^*_{\text{fair}}$ and consequently

$$d(s,c') \leq d(s,c) + d(c,c') \leq r^*_{\text{fair}} + (2^{j+1} + 2^j - 2) r^*_{\text{fair}} = (2^{j+1} + 2^j - 1) r^*_{\text{fair}}.$$

Hence,

$$\min_{c \in C_0 \cup \bigcup_{l=1}^{j} C_l} d(s,c) \leq (2^{j+1} + 2^j - 1) r^*_{\text{fair}}, \quad s \in S_a^{(j+1)}. \tag{12}$$

For every $s \in S_b^{(j+1)}$ there exists $c \in C^*_{\text{fair}} \cap \bigcup_{W \in \mathcal{G}^{(j)}} W$ with $d(s,c) \leq r^*_{\text{fair}}$. Let $C^*_{\text{fair}} \cap \bigcup_{W \in \mathcal{G}^{(j)}} W = \{\tilde{c}_1^*, \ldots, \tilde{c}_{\tilde{k}}^*\}$ with $\tilde{k} = \sum_{W \in \mathcal{G}^{(j)}} k_W$, where k_W is the number of requested centers from group W. We can write

$$S_b^{(j+1)} = \bigcup_{l=1}^{\tilde{k}} \left\{ s \in S_b^{(j+1)} : d(s, \tilde{c}_l^*) \leq r^*_{\text{fair}} \right\},$$

where some of the sets in this union might be empty, but that does not matter. Note that for every $l = 1, \ldots, \tilde{k}$ we have

$$d(s, s') \leq 2r^*_{\text{fair}}, \quad s, s' \in \left\{ s \in S_b^{(j+1)} : d(s, \tilde{c}^*_l) \leq r^*_{\text{fair}} \right\} \tag{13}$$

due to the triangle inequality. It is

$$S^{(j+1)} = S_a^{(j+1)} \cup S_b^{(j+1)} = S_a^{(j+1)} \cup \bigcup_{l=1}^{\tilde{k}} \left\{ s \in S_b^{(j+1)} : d(s, \tilde{c}^*_l) \leq r^*_{\text{fair}} \right\}$$

and when, in Line 3 of Algorithm 4, we run Algorithm 1 on $S^{(j+1)}$ with $k = \tilde{k}$ and initial centers $C_0 \cup \bigcup_{l=1}^{j} C_l$, one of the following three cases has to happen (we denote the centers returned by Algorithm 1 in this $(j+1)$-th call of Algorithm 4 by $\widetilde{F}^A = \{\tilde{f}_1^A, \ldots, \tilde{f}_{\tilde{k}}^A\}$ and assume that for $1 \leq l < l' \leq \tilde{k}$ Algorithm 1 has chosen $\tilde{f}_l^A$ before $\tilde{f}_{l'}^A$):

- For every $l \in \{1, \ldots, \tilde{k}\}$ there exists $l' \in \{1, \ldots, \tilde{k}\}$ such that $\tilde{f}_{l'}^A \in \{s \in S_b^{(j+1)} : d(s, \tilde{c}^*_l) \leq r^*_{\text{fair}}\}$. In this case it immediately follows that

$$\min_{c \in \widetilde{F}^A} d(s, c) \leq 2r^*_{\text{fair}}, \quad s \in S_b^{(j+1)},$$

  and using (12) we obtain

$$\min_{c \in C_0 \cup \bigcup_{l=1}^{j} C_l \cup \widetilde{F}^A} d(s, c) \leq (2^{j+1} + 2^j - 1) r^*_{\text{fair}}, \quad s \in S^{(j+1)}.$$

- There exists $l' \in \{1, \ldots, \tilde{k}\}$ such that $\tilde{f}_{l'}^A \in S_a^{(j+1)}$. When Algorithm 1 picks $\tilde{f}_{l'}^A$, any other element in $S^{(j+1)}$ cannot be at a larger minimum distance from a center in $C_0 \cup \bigcup_{l=1}^{j} C_l$ or an already chosen center in $\{\tilde{f}_{l'}^A, \ldots, \tilde{f}_{l'-1}^A\}$ than $\tilde{f}_{l'}^A$. It follows from (12) that

$$\min_{c \in C_0 \cup \bigcup_{l=1}^{j} C_l \cup \widetilde{F}^A} d(s, c) \leq (2^{j+1} + 2^j - 1) r^*_{\text{fair}}, \quad s \in S^{(j+1)}.$$

- There exist $l \in \{1, \ldots, \tilde{k}\}$ and $l', l'' \in \{1, \ldots, \tilde{k}\}$ with $l' < l''$ such that $\tilde{f}_{l'}^A, \tilde{f}_{l''}^A \in \{s \in S_b^{(j+1)} : d(s, \tilde{c}^*_l) \leq r^*_{\text{fair}}\}$. When Algorithm 1 picks $\tilde{f}_{l''}^A$, any other element in $S^{(j+1)}$ cannot be at a larger minimum distance from a center in $C_0 \cup \bigcup_{l=1}^{j} C_l$ or an already chosen center in $\{\tilde{f}_{l'}^A, \ldots, \tilde{f}_{l''-1}^A\}$ than $\tilde{f}_{l''}^A$. Because of $d(\tilde{f}_{l'}^A, \tilde{f}_{l''}^A) \leq 2r^*_{\text{fair}}$ according to (13), it follows that

$$\min_{c \in C_0 \cup \bigcup_{l=1}^{j} C_l \cup \widetilde{F}^A} d(s, c) \leq 2r^*_{\text{fair}} \leq (2^{j+1} + 2^j - 1) r^*_{\text{fair}}, \quad s \in S^{(j+1)}.$$

In any case, we have

$$\min_{c \in C_0 \cup \bigcup_{l=1}^{j} C_l \cup \widetilde{F}^A} d(s, c) \leq (2^{j+1} + 2^j - 1) r^*_{\text{fair}}, \quad s \in S^{(j+1)}. \tag{14}$$

Similarly to the base case, it follows from the triangle inequality that after running Algorithm 3 in Line 8 of Algorithm 4 and exchanging some of the centers in $\widetilde{F}^A$, we have

$$d(s, c(s)) \leq 2(2^{j+1} + 2^j - 1) r^*_{\text{fair}} = (2^{j+2} + 2^{j+1} - 2) r^*_{\text{fair}}$$

for every $s \in S^{(j+1)}$, where $c(s)$ denotes the center of its cluster. In particular, we have

$$\min_{c \in C_0 \cup \bigcup_{l=1}^{j+1} C_l} d(s, c) \leq (2^{j+2} + 2^{j+1} - 2) r^*_{\text{fair}}, \quad s \in \left(S^{(j+1)} \setminus S^{(j+2)} \right) \cup \left(C_0 \cup \bigcup_{l=1}^{j+1} C_l \right),$$

and this completes the proof of (11).

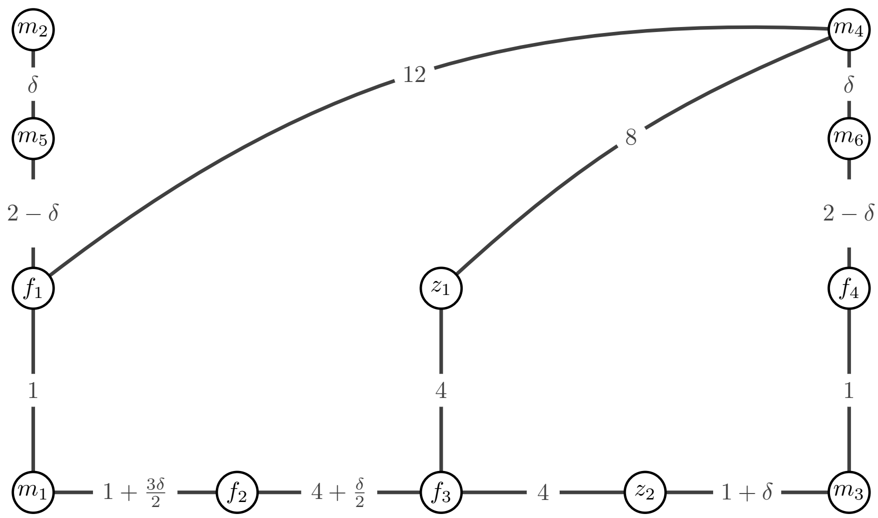

Figure 8. An example showing that Algorithm 4 is not a $(8-\varepsilon)$-approximation algorithm for any $\varepsilon > 0$.

If in the T-th call of Algorithm 4 the algorithm terminates from Line 10, it follows from (11) that

$$\min_{c \in C_0 \cup \bigcup_{l=1}^{T} C_l} d(s,c) \leq (2^{T+1} + 2^T - 2) r^*_{\text{fair}}, \quad s \in S. \tag{15}$$

In this case, since $T < m$, we have

$$2^{T+1} + 2^T - 2 \leq 2^m + 2^{m-1} - 2 < 2^m + 2^{m-1} - 1,$$

and (15) implies (10). If in the T-th call of Algorithm 4 the algorithm does not terminate from Line 10, it must terminate from Line 5. It follows from (11) that

$$\min_{c \in C_0 \cup \bigcup_{l=1}^{T-1} C_l} d(s,c) \leq (2^T + 2^{T-1} - 2) r^*_{\text{fair}}, \quad s \in \left(S \setminus S^{(T)}\right) \cup \left(C_0 \cup \bigcup_{l=1}^{T-1} C_l\right). \tag{16}$$

In the same way as we have shown (14) in the inductive step in the proof of (11), we can show that

$$\min_{c \in C_0 \cup \bigcup_{l=1}^{T-1} C_l \cup \widetilde{H}^A} d(s,c) \leq (2^T + 2^{T-1} - 1) r^*_{\text{fair}} \leq (2^m + 2^{m-1} - 1) r^*_{\text{fair}}, \quad s \in S^{(T)}, \tag{17}$$

where $\widetilde{H}^A$ is the set of centers returned by Algorithm 1 in the T-th call of Algorithm 4. Since $\bigcup_{l=1}^{T-1} C_l \cup \widetilde{H}^A$ is contained in the output C^A of Algorithm 4, (17) together with (16) implies (10).

Since running Algorithm 4 involves at most m (recursive) calls of the algorithm and the running time of each of these calls is dominated by the running times of Algorithm 1 and Algorithm 3, it follows that the running time of Algorithm 4 is $\mathcal{O}((|C_0|m + km^2)|S| + km^4)$. $\square$

Proof of Lemma 3:

Consider the example given by the weighted graph shown in Figure 8, where $0 < \delta < \frac{1}{10}$. We have $S = S_1 \dot{\cup} S_2 \dot{\cup} S_3$ with $S_1 = \{m_1, m_2, m_3, m_4, m_5, m_6\}$, $S_2 = \{f_1, f_2, f_3, f_4\}$ and $S_3 = \{z_1, z_2\}$. All distances are shortest-path-distances. Let $k_{S_1} = 4$, $k_{S_2} = 1$, $k_{S_3} = 1$ and $C_0 = \emptyset$. We assume that Algorithm 1 in Line 3 of Algorithm 4 picks f_1 as first center. It then chooses f_4 as second center, z_1 as third center, f_3 as fourth center, f_2 as fifth center and z_2 as sixth center. Hence, $\widetilde{C}^A = \{f_1, f_4, z_1, f_3, f_2, z_2\}$ and the corresponding clusters are $\{f_1, m_1, m_2, m_5\}$, $\{f_4, m_3, m_4, m_6\}$, $\{z_1\}$, $\{f_3\}$, $\{f_2\}$ and $\{z_2\}$. When running Algorithm 3 in Line 8 of Algorithm 4, it replaces f_1 with one of m_1, m_2 or m_5 and it replaces f_4

with one of m_3, m_4 or m_6. Assume that it replaces f_1 with m_2 and f_4 with m_4. Algorithm 3 then returns $\mathcal{G} = \{S_2, S_3\}$ and when recursively calling Algorithm 4 in Line 12, we have $S' = \{f_2, f_3, z_1, z_2\}$ and $C' = \{m_2, m_4\}$. In the recursive call, the given centers are C' and Algorithm 1 chooses f_3 and f_2. The corresponding clusters are $\{f_3, z_1, z_2\}$, $\{f_2\}$, $\{m_2\}$ and $\{m_4\}$. When running Algorithm 3 with clusters $\{f_3, z_1, z_2\}$ and $\{f_2\}$, it replaces f_3 with either z_1 or z_2 and returns $\mathcal{G} = \emptyset$, that is afterwards we are done. Assume Algorithm 3 replaces f_3 with z_2. Then the centers returned by Algorithm 4 are z_2, f_2, m_2, m_4 and two arbitrary elements from S_1, which we assume to be m_5 and m_6. These centers have a cost of 8 (incurred for z_1). However, an optimal solution such as $C^*_{\text{fair}} = \{m_1, m_2, m_3, m_4, f_3, z_1\}$ has cost only $1 + \frac{3\delta}{2}$. Choosing δ sufficiently small shows that Algorithm 4 is not a $(8 - \varepsilon)$-approximation algorithm for any $\varepsilon > 0$. □

B. Further Experiments

In Figure 9 we show the costs of the approximate solutions produced by our algorithm (Alg. 4) and the algorithm by Chen et al. (2016) (M.C.) in the run-time experiment shown in the right part of Figure 3. In Figure 10, Figure 11 and Figure 12 we provide similar experiments as shown in Figure 6, Figure 2 and Figure 5, respectively.

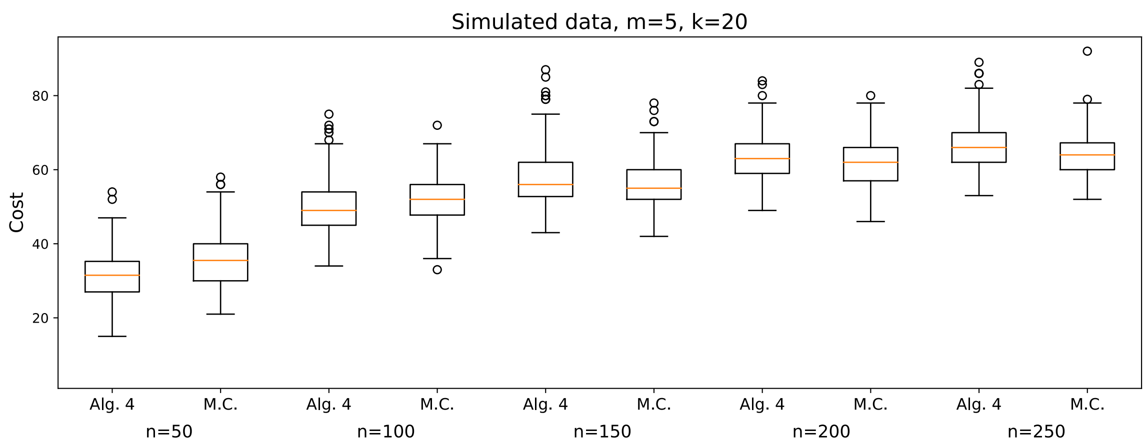

Figure 9. Cost of the output of our algorithm (Alg. 4) in comparison to the algorithm by Chen et al. (M.C.) in the run-time experiment shown in the right part of Figure 3.

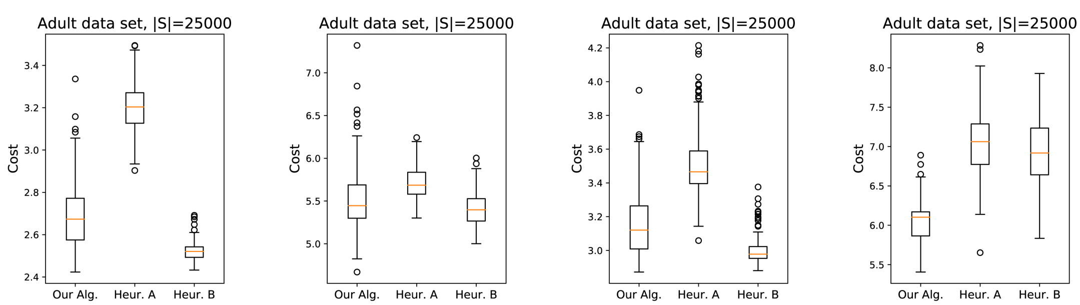

Figure 10. Similar experiments on the Adult data set as shown in Figure 6, but with different values of k_{S_i}. **1st plot:** $m = 2$, $k_{S_1} = 300$, $k_{S_2} = 100$ (S_1 corresponds to male and S_2 to female). **2nd plot:** $m = 2$, $k_{S_1} = k_{S_2} = 25$. **3rd plot:** $m = 5$, $k_{S_1} = 214$, $k_{S_2} = 8$, $k_{S_3} = 2$, $k_{S_4} = 2$, $k_{S_5} = 24$ ($S_1 \sim$ White, $S_2 \sim$ Asian-Pac-Islander, $S_3 \sim$ Amer-Indian-Eskimo, $S_4 \sim$ Other, $S_5 \sim$ Black). **4th plot:** $m = 5$, $k_{S_1} = k_{S_2} = k_{S_3} = k_{S_4} = k_{S_5} = 10$.

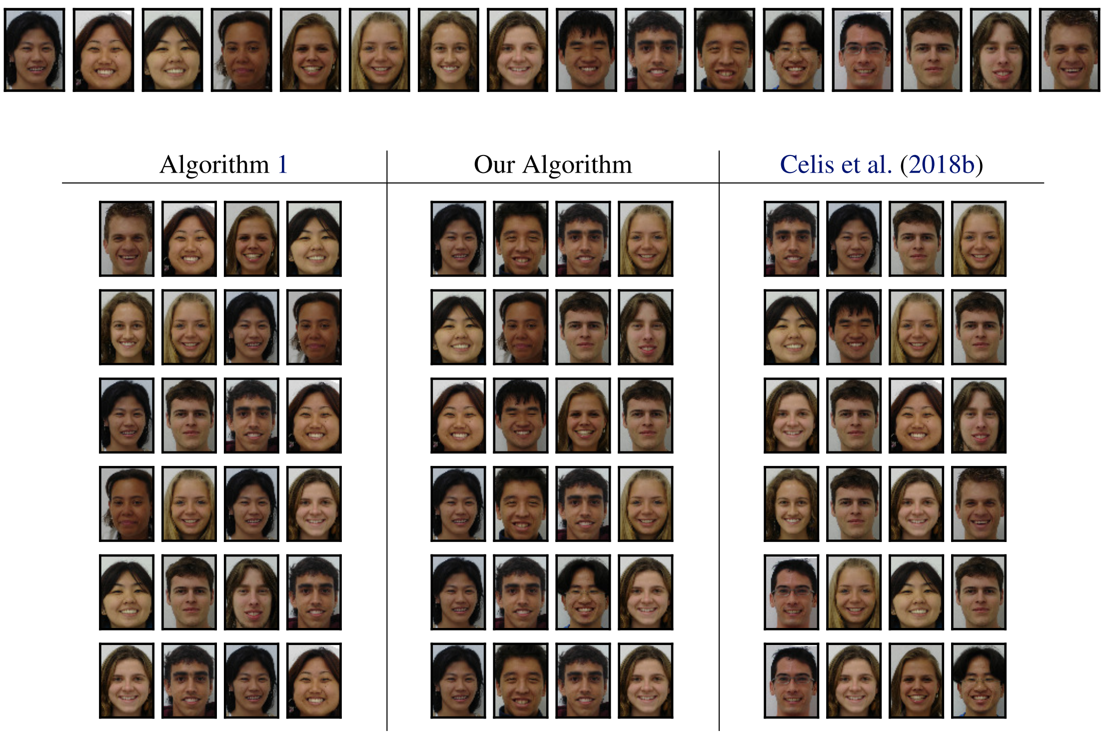

Figure 11. Similar experiment as shown in Figure 2. A data set consisting of 16 images of faces (8 female, 8 male) and six summaries computed by the unfair Algorithm 1, our algorithm and the algorithm of Celis et al. (2018b). The images are taken from the FEI face database available on `https://fei.edu.br/~cet/facedatabase.html`. Note that in this experiment (and the one shown in Figure 2) we are dealing with a very small number of images solely for the purpose of easy visual digestion.

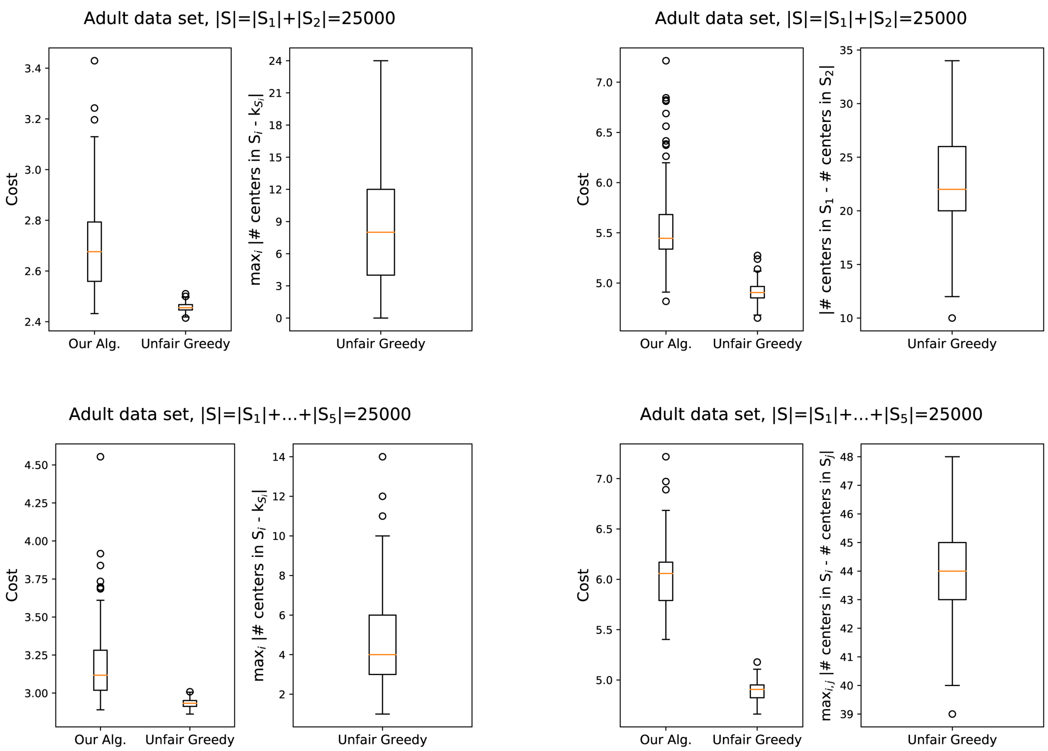

Figure 12. Similar experiments on the Adult data set as shown in Figure 5, but with different values of k_{S_i}. **Top left:** $m = 2$, $k_{S_1} = 300$, $k_{S_2} = 100$ (S_1 corresponds to male and S_2 to female). **Top right:** $m = 2$, $k_{S_1} = k_{S_2} = 25$. **Bottom left:** $m = 5$, $k_{S_1} = 214$, $k_{S_2} = 8$, $k_{S_3} = 2$, $k_{S_4} = 2$, $k_{S_5} = 24$ ($S_1 \sim$ White, $S_2 \sim$ Asian-Pac-Islander, $S_3 \sim$ Amer-Indian-Eskimo, $S_4 \sim$ Other, $S_5 \sim$ Black). **Bottom right:** $m = 5$, $k_{S_1} = k_{S_2} = k_{S_3} = k_{S_4} = k_{S_5} = 10$.