A. Improved analysis for the GreeDI algorithm with an arbitrary partition

Let OPT be an arbitrary collection of k elements from V, and let M be the set of machines that have some element of OPT placed on them. For each $j \in M$ let O_j be the set of elements of OPT placed on machine j, and let $r_j = |O_j|$ (note that $\sum_{j \in M} r_j = k$). Similarly, let S_j be the set of elements returned by the greedy algorithm on machine j. Let $e_j^i \in S_j$ denote the element chosen in the ith round of the greedy algorithm on machine j, and let S_j^i denote the set of all elements chosen in rounds 1 through i. Finally, let $S = \cup_{j \in M} S_j$ and $S^i = \cup_j S_j^i$.

In the following, we use $f_A(B)$ to denote $f(A \cup B) - f(A)$. We consider the marginal values:

$$x_j^i = f_{S_j^{i-1}}(e_j^i) = f(S_j^i) - f(S_j^{i-1})$$
$$y_j^i = f_{S_j^{i-1}}(O_j) = f(S_j^{i-1} \cup O_j) - f(O_j),$$

for each $1 \leq i \leq k$. Additionally, it will be convenient to define $x_j^{k+1} = y_j^{k+1} = 0$ and $S_j^{k+1} = S_j^k$ for all $j \in M$.

Because the elements e_j^i are selected greedily on each machine, the sequence $x_j^1, \ldots, x_j^k$ is non-increasing for all $j \in M$. Furthermore, we note that because each element e_j^i was selected by in the ith round of the greedy algorithm on machine j, we must have

$$x_j^i \geq \max_{o \in O_j \setminus S_j^{i-1}} f_{S_j^{i-1}}(o)$$

for all $j \in M$ and $i \in [k]$. Additionally, by submodularity, we have:

$$\begin{aligned} y_j^i &= f(S_j^{i-1} \cup O_j) - f(O_j) \\ &\leq \sum_{o \in O_j \setminus S_j^{i-1}} f_{S_j^{i-1}}(o) \\ &\leq r_j \cdot \max_{o \in O_j \setminus S_j^{i-1}} f_{S_j}^{i-1}(o). \end{aligned}$$

Therefore,

$$y_j^i \leq r_j \cdot x_j^i \tag{10}$$

for all $j \in M$ and $i \in [k]$.

We want to show that the set of elements S placed on the final machine contain a solution that is relatively good compared to OPT. We begin by proving the following lemma, which relates the value of $f(\mathrm{OPT})$ to the total value of the elements from the ith partial solutions produced on each of the machines.

Lemma 11. *For every $i \in [k]$ and every machine $j \in M$,*

$$f(\mathrm{OPT}) \leq f(S^i) + \sum_{j \in M} f_{S_j^i}(O_j).$$

Proof. We have

$$\begin{aligned} f(\mathrm{OPT}) &\leq f(\mathrm{OPT} \cup S^i) \\ &= f(S^i) + f_{S^i}(\mathrm{OPT}) \\ &\leq f(S^i) + \sum_{j \in M} f_{S^i}(O_j) \\ &\leq f(S^i) + \sum_{j \in M} f_{S_j^i}(O_j), \end{aligned}$$

where the first inequality follows from monotonicity of f, and the last two from submodularity of f. □

In order to obtain a bound on $f(\mathrm{OPT})$, it suffices to upper bound each term on the right hand side of the inequality from Lemma 11. We proceed step by step, according to the following intuition: if in all steps i the gain $f(S^i) - f(S^{i-1})$ is small compared to $\sum_{j \in M} x_j^i$, then we can use Lemma 11 and (10) to argue that $f(\mathrm{OPT})$ must also be relatively small. On the other hand, if $f(S^i) - f(S^{i-1})$ is large compared to $\sum_{j \in M} x_j^i$, then $S^i \setminus S^{i-1}$ is a reasonably good solution that is available on the final machine.

We proceed by balancing these two cases for a particular critical step i. Specifically, fix $i \leq k$ be the smallest value such that:

$$\sum_{j \in M} r_j \cdot x_j^{i+1} \leq \sqrt{k} \cdot \left[f(S^{i+1}) - f(S^i) \right]. \tag{11}$$

Note that some such value i must exist, since for $i = k$, both sides of (11) are equal to zero. We now derive a bound on each term on the right of Lemma 11. Let $\tilde{\mathrm{OPT}} \subseteq S$ be a set of k elements from S that maximizes f.

Lemma 12. $f(S^i) \leq \sqrt{k} \cdot f(\tilde{\mathrm{OPT}})$.

Proof. Because i is the smallest value for which (11) holds, we must have

$$\sum_{j \in M} r_j \cdot x_j^\ell > \sqrt{k} \cdot \left[f(S^\ell) - f(S^{\ell-1}) \right], \text{ for all } \ell \leq i.$$

Therefore,

$$\begin{aligned} \sum_{j \in M} r_j \cdot f(S_j^i) &= \sum_{j \in M} \sum_{\ell=1}^{i} r_j \cdot \left[f(S_j^\ell) - f(S_j^{\ell-1}) \right] \\ &= \sum_{j \in M} \sum_{\ell=1}^{i} r_j \cdot x_j^\ell \\ &= \sum_{\ell=1}^{i} \sum_{j \in M} r_j \cdot x_j^\ell \\ &> \sum_{\ell=1}^{i} \sqrt{k} \cdot \left[f(S^\ell) - f(S^{\ell-1}) \right] \end{aligned}$$

$$= \sqrt{k} \cdot f(S^i),$$

and so,

$$\begin{aligned} f(S^i) &< \frac{1}{\sqrt{k}} \sum_{j \in M} r_j \cdot f(S^i_j) \\ &\leq \frac{1}{\sqrt{k}} \sum_{j \in M} r_j \cdot f(S_j) && \text{(By monotonicity)} \\ &\leq \frac{1}{\sqrt{k}} \sum_{j \in M} r_j \cdot f(\tilde{\mathrm{OPT}}) && (S_j \subseteq S \text{ is feasible}) \\ &= \sqrt{k} \cdot f(\tilde{\mathrm{OPT}}). \qquad \square \end{aligned}$$

Lemma 13. $\sum_{j \in M} f_{S^i_j}(O_j) \leq \sqrt{k} \cdot f(\tilde{\mathrm{OPT}})$.

Proof. We consider two cases:

Case: $i < k$**.** We have $i + 1 \leq k$, and by (10) we have $f_{S^i_j}(O_j) = y^{i+1}_j \leq r_j \cdot x^{i+1}_j$ for every machine j. Therefore:

$$\begin{aligned} \sum_{j \in M} f_{S^i_j}(O_j) &\leq \sum_{j \in M} r_j \cdot x^{i+1}_j \\ &\leq \sqrt{k} \cdot (f(S^{i+1}) - f(S^i)) \\ & \qquad\qquad \text{(By definition of } i) \\ &\leq \sqrt{k} \cdot f(S^{i+1} \setminus S^i) \quad \text{(By submodularity)} \\ &\leq \sqrt{k} \cdot f(\tilde{\mathrm{OPT}}), \end{aligned}$$

where the final line follows from the fact that $|S^{i+1} \setminus S^i| \leq k$ and so $S^{i+1} \setminus S^i$ is a feasible solution.

Case: $i = k$**.** By submodularity of f and (10), we have

$$f_{S^i_j}(O_j) = f_{S^k_j}(O_j) \leq f_{S^{k-1}_j}(O_j) = y^k_j \leq r_j \cdot x^k_j.$$

Moreover, since the sequence $x^1_j, \ldots, x^k_j$ is non-increasing for all j,

$$x^k_j \leq \frac{1}{k} \sum_{\ell=1}^{k} x^\ell_j = \frac{1}{k} \cdot f(S_j).$$

Therefore,

$$\begin{aligned} \sum_{j \in M} f_{S^i_j}(O_j) &\leq \sum_{j \in M} r_j \cdot x^k_j \\ &\leq \sum_{j \in M} \frac{r_j}{k} \cdot f(S_j) \\ &\leq \sum_{j \in M} \frac{r_j}{k} \cdot f(\tilde{\mathrm{OPT}}) \quad (S_j \subseteq S \text{ is feasible}) \\ &= f(\tilde{\mathrm{OPT}}). \end{aligned}$$

Thus, in both cases, we have $\sum_{j \in M} f_{S^i_j}(O_j) \leq \sqrt{k} \cdot f(\tilde{\mathrm{OPT}})$ as required. $\square$

Our main theorem then follows directly from Lemmas 11, 12, and 13:

Theorem 14. $f(\mathrm{OPT}) \leq 2\sqrt{k} f(\tilde{\mathrm{OPT}})$.

Because the standard greedy algorithm executed on the last machine is a $(1 - 1/e)$-approximation, we have the following corollary.

Corollary 15. *The distributed greedy algorithm gives a* $\frac{(1-1/e)}{2\sqrt{k}}$ *approximation for maximizing a monotone submodular function subject to a cardinality constraint k, regardless of how the elements are distributed.*

B. A tight example for the GreeDI algorithm with an arbitrary partition

Here we give a family of examples that show that the GreeDI algorithm of Mirzasoleiman *et al.* cannot achieve an approximation better than $1/\sqrt{k}$ if the partition of the elements onto the machines is arbitrary.

Consider the following instance of Max k-Coverage. We have $\ell^2 + 1$ machines and $k = \ell + \ell^2$. Let N be a ground set with $\ell^2 + \ell^3$ elements, $N = \{1, 2, \ldots, \ell^2 + \ell^3\}$. We define a coverage function on a collection $\mathcal{S}$ of subsets of N as follows. In the following, we define how the sets of $\mathcal{S}$ are partitioned on the machines.

On machine 1, we have the following ℓ sets from OPT: $O_1 = \{1, 2, \ldots, \ell\}$, $O_2 = \{\ell + 1, \ldots, 2\ell\}$, $\ldots$, $O_\ell = \{\ell^2 - \ell + 1, \ldots, \ell^2\}$. We also pad the machine with copies of the empty set.

On machine $i > 1$, we have the following sets. There is a single set from OPT, namely

$$O'_i = \{\ell^2 + (i-1)\ell + 1, \ell^2 + (i-1)\ell + 2, \ldots, \ell^2 + i\ell\}.$$

Additionally, we have ℓ sets that are designed to fool the greedy algorithm; the j-th such set is $O_j \cup \{\ell^2 + (i-1)\ell + j\}$. As before, we pad the machine with copies of the empty set.

The optimal solution is $O_1, \ldots, O_\ell, O'_1, \ldots, O'_{\ell^2}$ and it has a total coverage of $\ell^2 + \ell^3$.

On the first machine, Greedy picks the ℓ sets $O_1, \ldots, O_m$ from OPT and ℓ^2 copies of the empty set. On each machine $i > 1$, Greedy first picks the ℓ sets $A_j = O_j \cup \{\ell^2 + (i-1)\ell + j\}$, since each of them has marginal value greater than O'_i. Once Greedy has picked all of the A_j's, the marginal value of O'_i becomes zero and we may assume that Greedy always picks the empty sets instead of O'_i.

Now consider the final round of the algorithm where we run Greedy on the union of the solutions from each of the machines. In this round, regardless of the algorithm, the sets picked can only cover $\left\{1, \ldots, \ell^2\right\}$ (using the set $O_1, \ldots, O_\ell$) and one additional item per set for a total of $2\ell^2$ elements. Thus the total coverage of the final solution is at most $2\ell^2$. Hence the approximation is at most $\frac{2\ell^2}{\ell^2+\ell^3} = \frac{2}{1+\ell} \approx \frac{1}{\sqrt{k}}$.

C. The algorithm of Wolsey for non-monotone functions

In this section, we consider the algorithm of Wolsey (1982) for submodular maximization subject to a knapsack constraint. Let V denote the set of items. Let $w_i \in \mathbb{Z}_{\geq 0}$ denote the weight of item i. Let $b \in \mathbb{Z}_{\geq 0}$ be the capacity of the knapsack and $f : 2^V \to \mathbb{R}_{\geq 0}$ be a submodular function satisfying $f(\emptyset) = 0$. We wish to solve the problem:

$$\max\{f(S) : S \subseteq V, w(S) \leq b\},$$

where $w(S) = \sum_{i \in S} w_i$ is the total weight of the items in S. We emphasize that the function f is not necessarily monotone.

Wolsey's algorithm works exactly as the standard greedy algorithm shown in Algorithm 1, with two modifications: (1) at each step it takes the element i with highest non-negative *marginal profit density* $\theta_S(i) = \frac{f(S \cup \{i\}) - f(S)}{w_i}$, and (2) it returns either the greedy solution S or the best singleton solution $\{e\}$, whichever has the higher function value.

It is easily verified that the Lemma 2 holds for the resulting algorithm. In the following, we show that the algorithm satisfies the property (GP) with $\gamma = \frac{1}{3}$. More precisely, we will show that

$$f(T) \geq \frac{1}{3} f(T \cup O),$$

where T is the solution constructed by Wolsey's algorithm, and $O \subseteq V$ is any feasible solution.

Let S denote the Greedy solution, let $\{e\}$ denote the best singleton solution; the solution T is the better of the two solutions S and $\{e\}$. Let

$$j = \arg\max_{i \in O \setminus S} \theta_S(i).$$

We have

$$\begin{aligned}
f(S \cup O) &\leq f(S) + \sum_{i \in O \setminus S} (f(S \cup \{i\}) - f(S)) \\
&= f(S) + \sum_{i \in O \setminus S} w_i \theta_S(i) \\
&\leq f(S) + \sum_{i \in O \setminus S} w_i \theta_S(j) \\
&\leq f(S) + b \cdot \theta_S(j),
\end{aligned}$$

where the inequality on the first line follows from submodularity of f, the inequality on the third line from the definition of j, and the inequality on the last line from the fact that O (and hence $O \setminus S$) is feasible.

Thus, in order to complete the proof, it suffices to show that $b \cdot \theta_S(j) \leq 2 \max\{f(S), f(\{e\})\}$. We consider two cases based on the weight of j.

Suppose that $w_j > b/2$. We have

$$\begin{aligned}
b \cdot \theta_S(j) &< 2w_j \cdot \theta_S(j) \\
&= 2(f(S \cup \{j\}) - f(S)) \\
&\leq 2f(\{j\}) \leq 2f(\{e\}),
\end{aligned}$$

as desired.

Therefore we may assume that $w_j \leq b/2$. Let e_i denote the i-th element selected by the Greedy algorithm and let $S^i = \{e_1, e_2, \ldots, e_i\}$. Note that we may assume that $\theta_S(j) \geq 0$, since otherwise we would be done. Thus $\theta_{S^i}(j) \geq \theta_S(j) \geq 0$ for all i.

Let t be the largest index such that $w(S^t) \leq b - w_j$; note that $t < |S|$, since otherwise $S \cup \{j\}$ is a feasible solution with value greater than $f(S)$, which is a contradiction. We have $w(S^{t+1}) > b - w_j \geq b/2$.

In each iteration $i \leq t$, it was feasible to add j to the current solution; since the Greedy algorithm did not pick j, we must have $\theta_{S^{i-1}}(e_i) \geq \theta_{S^{i-1}}(j)$.

Finally, $f(S) \geq f(S^{t+1})$, since the Greedy algorithm only adds elements with non-negative marginal value. Therefore we have

$$\begin{aligned}
f(S) &\geq f(S^{t+1}) \\
&= \sum_{i=1}^{t+1} (f(S^i) - f(S^{i-1})) \\
&= \sum_{i=1}^{t+1} w_{e_i} \theta_{S^{i-1}}(e_i) \\
&\geq \sum_{i=1}^{t+1} w_{e_i} \theta_{S^{i-1}}(j) \\
&\geq \sum_{i=1}^{t+1} w_{e_i} \theta_S(j) \\
&= w(S^{t+1}) \cdot \theta_S(j) \\
&\geq \frac{b}{2} \cdot \theta_S(j).
\end{aligned}$$

Thus $b \cdot \theta_S(j) \leq 2f(S)$, as desired.