# 1 Approximate Supply Polyhedra for $Q||C_{min}$

Let the instance $\mathcal{I}$ of $Q||C_{min}$ have $m$ machines $M$ with demands $D_1 \geq \cdots \geq D_m$ and $n$ types of jobs $J$ with capacities $c_1 \geq \cdots \geq c_n$. A supply vector $(s_1, \ldots, s_n)$ indicates the number of jobs of each type available; a supply vector is feasible if together they can satisfy all the demands. We wish to find a convex set/polyhedra which captures all the feasible supply vectors. In particular, any feasible supply vector should be in the set, and given any (integer) supply vector in the set there should be an allocation which satisfies the demands to an $\alpha$-factor.

A feasible supply vector $(s_1, \ldots, s_n)$ must lie in the following polytope.

$$\mathcal{P}_{\mathsf{ass}} = \{(s_1, \ldots, s_n):$$

$$\forall j \in J, \quad \textstyle\sum_{i \in M} z_{ij} \leq s_j \tag{1}$$

$$\forall i \in M, \quad \textstyle\sum_{j \in C} z_{ij} \min(c_j, D_i) \geq D_i \tag{2}$$

$$\forall i \in M, j \in J, \quad z_{ij} \geq 0\} \tag{3}$$

Not all integral $(s_1, \ldots, s_n) \in \mathcal{P}_{\mathsf{ass}}$ need be feasible; but the following theorem shows given such a supply vector, there exists an assignment satisfying the demands up to a factor 2.

**Theorem 1.1.** *Given $(s_1, \ldots, s_n) \in \mathcal{P}_{\mathsf{ass}}$, there is an of assignment $\phi$ of the $s_j$ jobs of capacity $c_j$ to the machines such that for all $i \in M$, $\sum_{j:\phi(j)=i} c_j \geq D_i/2$.*

*Proof.* For simplicity, given the supply vector, abusing notation let $J$ denote the multiset of jobs where job $j$ appears $s_j$ times. We know that the LP(1)-(3) is feasible with the $s_j$ replaced by 1. Let $N = \sum_j s_j$.

The algorithm is a very simple greedy algorithm which doesn't look at the LP solution. Order the jobs (with multiplicities) in decreasing order of capacities $c_1 \geq c_2 \geq \cdots \geq c_N$, and order the machines in decreasing order of $D_i$'s, that is, $D_1 \geq D_2 \geq \ldots \geq D_m$. Starting with machine $i = 1$ and job $j = 1$, assign jobs $j$ to $i$ if the total capacity filled in machine $i$ is $< D_i/2$ and move to the next job. Otherwise, call machine $i$ happy and move to the next machine. Obviously, if all machines are happy at the end we have found our assignment.

The non-trivial part is to prove that if some machine is unhappy, then the LP(1)-(3) is infeasible (with $s_j$ replaced by 1). To do so, we take the Farkas dual of the LP; the following LP is feasible iff LP(1)-(3) is infeasible.

$$\textstyle\sum_{i=1}^{m} \beta_i D_i > \sum_{j=1}^{n} \alpha_j \tag{4}$$

$$\forall i \in M, j \in J \quad \beta_i \min(c_j, D_i) \leq \alpha_j \tag{5}$$

$$\forall i \in M, \quad \beta_i \geq 0 \tag{6}$$

Suppose machine $i^\star$ is the first machine which is unhappy. Let $S_1, \ldots, S_{i^\star-1}$ be the jobs assigned to machines 1 to $(i^\star - 1)$ and $S_i^\star$ be the remainder of jobs. We have $\sum_{j \in S_i^\star} c_j < D_i^\star/2$. We also have for all $1 \leq i \leq i^\star$, $\sum_{j \in S_i} \min(c_j, D_i) \leq D_i$. We now describe a feasible solution to (4)-(6).

Given the assignment $S_i$'s, call a machine $i$ *overloaded* if $S_i$ contains a single jobs $j_i$ with $c_{j_i} \geq D_i$. We let $\beta_1 = 1$. For $1 \leq i < i^\star$, we have the following three-pronged rule

- If $i+1$ is not overloaded, $\beta_{i+1} = \beta_i$.
- If $i+1$ is overloaded, and so is $i$, then $\beta_{i+1} = \beta_i \cdot D_i/D_{i+1}$.
- If $i+1$ is overloaded but $i$ is not, then $\beta_{i+1} = \beta_i \cdot c_{j_{i+1}}/D_{i+1}$, where $j_{i+1}$ is the job assigned to $i+1$.

For any job $j$ assigned to machine $i$, we set $\alpha_j = \beta_i \min(c_j, D_i)$. Since for any $S_i$, we have $\sum_{j \in S_i} \min(c_j, D_i) \leq D_i$ and $\sum_{j \in S_i^\star} c_j < D_i^\star/2$, the given $(\alpha, \beta)$ solution satisfies (4). We now prove that it satisfies (5). From the construction of the $\beta$'s the following claims follow.

**Claim 1.2.** $\beta_1 \leq \beta_2 \leq \cdots \leq \beta_m$.

**Claim 1.3.** $\beta_1 D_1 \geq \beta_2 D_2 \geq \cdots \geq \beta_m D_m$.

*Proof.* The only non-obvious case is if $i+1$ is overloaded but $i$ is not: in this case $\beta_{i+1} D_{i+1} = \beta_i c_{j_{i+1}}$. But since $i$ is not overloaded, let $j$ be some job assigned to $i$ with $c_j \leq D_i$. By the greedy rule, $c_j \geq c_{j_{i+1}}$, and so $\beta_{i+1} D_{i+1} \leq \beta_i D_i$. $\square$

Now fix a job $j$ and let $i$ be the machine it is assigned to. Note (5) holds for $(i,j)$ and we need to show (5) holds for all $(i',j)$ too. I don't see any more glamorous way than case analysis.

**Case 1:** $c_j \leq D_i$**.** In this case $\alpha_j = \beta_i c_j$ and $i$ is not overloaded. Let $i' < i$. Then we have $\beta_{i'} \min(c_j, D_{i'}) \leq \beta_{i'} c_j \leq \beta_i c_j$, where the last inequality follows from Claim 1.2.

Now let $i' > i$. If $c_j \leq D_{i'}$, then none of the machines from $i$ to $i'$ can be overloaded. Therefore, $\beta_{i'} = \beta_i$, and so $\beta_{i'} c_j = \beta_i c_j = \alpha_j$. So, we may assume $c_j > D_{i'}$ and we need to upper bound $\beta_{i'} D_{i'}$. Let $i'' > i$ be the first machine which is overloaded with job $j''$ say. By Claim 1.3, we have $\beta_{i'} D_{i'} \leq \beta_{i''} D_{i''}$. Now note that $\beta_{i''} D_{i''} = \beta_{i''-1} c_{j''} = \beta_i c_{j''} \leq \beta_i c_j = \alpha_j$ where the second equality follows since none of the machines from $i$ to $i''-1$ were overloaded.

**Case 2:** $c_j > D_i$**.** In this case $\alpha_j = \beta_i D_i$ and $i$ is overloaded. Let $i' > i$. Then, $\beta_{i'} \min(c_j, D_{i'}) = \beta_{i'} D_{i'} \leq \beta_i D_i$ where the last inequality follows from Claim 1.3.

Let $i' < i$. Let $i' \leq i'' < i$ be the smallest entry such that $c_j > D_{i''}$. Note that all machines from $i''$ to $i$ must be overloaded implying $\beta_{i''} D_{i''} = \beta_i D_i$. Since $c_j \leq D_{i'}$ (in case $i' < i''$), we need to upper bound $\beta_{i'} c_j$. By Claim 1.2, $\beta_{i'} c_j \leq \beta_{i''-1} c_j$. Now, if $i''-1$ were overloaded, then $\beta_{i''} D_{i''} = \beta_{i''-1} D_{i''-1} \geq \beta_{i''-1} c_j$ where the last inequality follows from definition of $i''$. Together, we get $\beta_{i'} c_j \leq \beta_i D_i$. $\square$

**Lemma 1.4.** *Suppose* $(y_1, \ldots, y_n) \in \mathcal{P}_{\mathsf{ass}}$*. Let* $(\bar{y}_1, \ldots, \bar{y}_n)$ *be a vector such that for all* $1 \leq i \leq n$*,* $\sum_{j \leq i} \bar{y}_j \geq \sum_{j \leq i} y_i$*. Then* $(\bar{y}_1, \ldots, \bar{y}_n) \in \mathcal{P}_{\mathsf{ass}}$*.*

*Proof.* By induction, let us assume the lemma is true for all $\bar{y}$ with $\bar{y}_1 = y_1$ which satisfies the prefix-sum condition. Let $\bar{\bar{y}}$ be the vector with $\bar{\bar{y}}_1 = y_1$, $\bar{\bar{y}}_2 = \bar{y}_2 + \bar{y}_1 - y_1$, and $\bar{\bar{y}}_i = \bar{y}_i$ otherwise. Since $\bar{\bar{y}} \in \mathcal{P}_{\mathsf{ass}}$, there is an assignment $z_{ij}$ satisfying (1)-(3) with $s_j = \bar{\bar{y}}_j$. We now describe a feasible solution $\bar{z}_{ij}$ with $s_j = \bar{y}_j$.

Let $\theta := \bar{y}_2 / \bar{\bar{y}}_2 \leq 1$ since $\bar{y}_1 \geq y_1$. Define $\bar{z}_{i2} = \theta z_{i2}$ for all $i$, and define $\bar{z}_{i1} = z_{i1} + (1-\theta) z_{i2}$. For $j = 2$, we have $\sum_{i \in M} \bar{z}_{i2} = \theta \sum_{i \in M} z_{i2} \leq \theta \bar{\bar{y}}_2 = \bar{y}_2$. For $j = 1$, we have $\sum_{i \in M} \bar{z}_{i1} = \sum_{i \in M} z_{i1} + (1-\theta) \sum_{i \in M} z_{i2} \leq \bar{\bar{y}}_1 + (1-\theta)\bar{\bar{y}}_2 = \bar{\bar{y}}_1 + \bar{\bar{y}}_2 - \bar{y}_2 = \bar{y}_1$. Since the other $z_{ij}$'s and $\bar{y}_j$'s are untouched, $\bar{z}_{ij}$ satisfies (1) with $\bar{y}_j$'s.

Now fix a machine $i$. The 'increase' in the LHS of (2) is $\sum_{j \in J} (\bar{z}_{ij} - z_{ij}) \min(c_j, D_i) = (1-\theta) z_{i2} \min(c_1, D_i) - (1-\theta) z_{i2} \min(c_2, D_i) \geq 0$ since $c_1 \geq c_2$.

$\square$