EC.2. Extended Online Appendix: Integrality Gap of a Related Maximization Problem

In this section, we obtain guarantees on the integrality gap of a related maximization problem, specifically, the maximization version of (12)-(18):

$$\max \sum_{d\in\mathcal{D}^2}\sum_{\ell\in\mathcal{L}(d)}\sum_{t\in\mathcal{T}_2}\tilde{c}_{d,t}z_{d,\ell,t} - \sum_{d\in\mathcal{D}^2}\sum_{s\in\mathcal{S}(d)} h_{d,s}w_{d,s} \tag{EC.1}$$

$$\text{s.t. constraints (13)-(18) hold,} \tag{EC.2}$$

where $\tilde{c}_{d,t} = u_d - c_{d,t}$, for all demands d and times t. One can think of the maximization objective as each demand d as contributing $\tilde{c}_{d,t}$ to the objective value if it is fulfilled on day t, while incurring a cost $h_{d,s}$ for shipping if supplier s is used. Obviously in the optimal solution, we always have $\tilde{c}_{d,t} - h_{d,s} \geq 0$ for the selected dock-date and supplier, otherwise a better value could be obtained by simply not fulfilling the demand. We show the following results for this formulation:

Theorem EC.1. *With only Property 1, the integrality gap of* (EC.1)-(EC.2) *is at least* $\frac{3}{2}$.

Proof of Theorem EC.1. We can show this result using the same construction as in the proof of Theorem 4. That example leads to optimal value of the LP relaxation of (EC.1)-(EC.2) of at least 1.5 while the optimal value of the IP formulation is 1; hence, the integrality gap is at least $\frac{3}{2}$.

We complement this result with an upper bound on the integrality gap.

Theorem EC.2. *With only Property 1, the integrality gap of* (EC.1)-(EC.2) *is at most 4.*

To prove this result, we start by solving the following LP relaxation for the problem:

$$\max \sum_{d\in\mathcal{D}^2}\sum_{\ell\in\mathcal{L}(d)}\sum_{t\in\mathcal{T}_2}\tilde{c}_{d,t}z_{d,\ell,t} - \sum_{d\in\mathcal{D}^2}\sum_{s\in\mathcal{S}(d)} h_{d,s}w_{d,s} \tag{EC.3}$$

$$\text{s.t.} \sum_{\ell\in\mathcal{L}(d)}\sum_{t\in\mathcal{T}_2} z_{d,\ell,t} \leq 1 \qquad \forall d\in\mathcal{D}^2 \tag{EC.4}$$

$$\sum_{\ell\in\mathcal{L}(d)}\sum_{t\in\mathcal{T}_2} z_{d,\ell,t} = \sum_{s\in\mathcal{S}(d)} w_{d,s} \qquad \forall d\in\mathcal{D}^2 \tag{EC.5}$$

$$\delta_{p,t,2} \geq \sum_{d\in\mathcal{D}^2} \sum_{\ell\in p\cap\mathcal{L}(d)} z_{d,\ell,t} \qquad \forall (p,t)\in\mathcal{H} : t\in\mathcal{T}_2 \tag{EC.6}$$

$$\rho_{\ell,2} \geq \sum_{d\in\mathcal{D}^2:\ell\in\mathcal{L}(d)} \sum_{t\in\mathcal{T}_2} z_{d,\ell,t} \qquad \forall \ell\in\mathcal{L} \tag{EC.7}$$

$$\sigma_{s,2} \geq \sum_{d\in\mathcal{D}^2} w_{d,s} \qquad \forall s\in\mathcal{S} \tag{EC.8}$$

$$0 \leq z_{d,\ell,t}, w_{d,s} \leq 1. \tag{EC.9}$$

Let z^f and w^f denote the optimal solution vectors of this LP relaxation (which are possibly fractional). The core of our proof is the design of a rounding algorithm that starts with this potentially fractional solution (z^f, w^f) for the LP relaxation and rounds it appropriately to obtain an integral solution with at least 1/4 of the objective value. The details follow.

Rounding Algorithm. A high-level overview of this algorithm is given in Algorithm 2. In each iteration, the algorithm selects a demand d^* that corresponds to at least one fractional decision variable and ensures that at the end of the iteration the solution for d^* is integral (i.e., exactly one dock-date, location, and supplier are selected). Details on the selection of d^* and the rounding process can be found in Algorithm 2.

Algorithm 2: Rounding Algorithm

while *there exists a fractional variable* **do**

- Let D^f denote the set of demands with at least one fractional variable z^f or w^f.
- For each $d\in D^f$ calculate $V_d = \max\{\tilde{c}_{d,t} - h_{d,s} \mid \forall d,t,s \text{ where } w_{d,s} > 0 \text{ and } \exists \ell : z_{d,\ell,t} > 0\}$.
- Select the demand with the best value: $d^* = \arg\max_{d\in D^f} V_d$.
- Let t^*, ℓ^*, s^* be indices such that $V_{d^*} = \tilde{c}_{d^*,t^*} - h_{d^*,s^*}$ and $z^f_{d^*,\ell^*,t^*} > 0$ and $w^f_{d^*,s^*} > 0$. By construction, at least one of the $z^f_{d^*,\ell^*,t^*}$ and $w^f_{d^*,s^*}$ is fractional.
- Round up $z^f_{d^*,\ell^*,t^*} := 1$ and $w^f_{d^*,s^*} := 1$.
- Decrease the remaining variables so that (EC.4)-(EC.8) continue to hold.

end

We now focus on the last step of the algorithm and provide additional details. Since at least one of the variables for d^* is increased by rounding up to 1, there is a possibility that some of the constraints (EC.4)-(EC.8) will be violated. To address this issue, we need to decrease appropriately some of the remaining variables. Algorithm 3 is a useful subroutine that will be evoked for constraints that are violated. We next discuss how that is handled for each constraint.

Algorithm 3: Variable Decrease Subroutine (set of variables V, value κ)

---

1 Sort the variables in V in an arbitrary order.

2 **while** $\kappa > 0$ **do**

3   Let x denote the first non-zero variable in V.

4   **if** $x < \kappa$ **then** $\kappa := \kappa - x$ and $x := 0$;

5   **else** $x := x - \kappa$ and $\kappa := 0$.

6 **end**

Constraints (EC.4): For the selected demand d^*, round down all fractional variables $z^f_{d^*,\ell,t}$, leaving $z^f_{d^*,\ell^*,t^*} = 1$ as the only non-zero z variable for the demand.

Constraints (EC.7): For the selected location ℓ^*, if the corresponding capacity constraint (EC.7) is violated, then let V denote the set of all fractional variables $z^f_{d,\ell^*,t}$ that participate in the violated constraint and let κ be the amount by which the constraint is violated; evoke Algorithm 3 with variable set V and value κ.

Constraints (EC.8): For the selected supplier s^*, if the corresponding supplier constraint (EC.8) is violated, then let V denote the set of all fractional variables w^f_{d,s^*} that participate in this constraint and κ the amount by which the constraint is violated, and evoke Algorithm 3 with this variable set V and value κ.

Constraints (EC.6): For the selected time t^* and location ℓ^*, find all related throughput constraints $(p, t^*) \in \mathcal{H}$ such that $\ell^* \in p$. From Property 1, these have a hierarchical structure. Order these sets from the smallest to the largest. For each such throughput set p', if the corresponding constraint for that throughput capacity is violated by some amount κ, then evoke Algorithm 3 with the value κ and the set V being the set of all fractional variables z^f_{d,ℓ,t^*} with $\ell \in p'$. Proceed similarly with the larger throughput sets $p \supset p'$ with $\ell^* \in p$ until no throughput constraint is violated.

Constraints (EC.5): To ensure all constraints (EC.5) are satisfied, we need to take multiple steps. First, for the selected demand d^*, round down all fractional variables $w^f_{d^*,s}$, leaving $w^f_{d^*,s^*} = 1$ the only non-zero w variable for the demand. Then, for each constraint (EC.5) that is violated due to the variable decreases to satisfy constraints (EC.6)-(EC.8):

- If $\sum_{\ell\in\mathcal{L}(d)}\sum_{t\in\mathcal{T}_2} z_{d,\ell,t} > \sum_{s\in\mathcal{S}(d)} w_{d,s}$, then evoke Algorithm 3 with $\kappa = \sum_{\ell\in\mathcal{L}(d)}\sum_{t\in\mathcal{T}_2} z_{d,\ell,t} - \sum_{s\in\mathcal{S}(d)} w_{d,s}$ and V the set of non-zero variables z associated with demand d.
- If $\sum_{\ell\in\mathcal{L}(d)}\sum_{t\in\mathcal{T}_2} z_{d,\ell,t} < \sum_{s\in\mathcal{S}(d)} w_{d,s}$, then evoke Algorithm 3 with $\kappa = \sum_{s\in\mathcal{S}(d)} w_{d,s} - \sum_{\ell\in\mathcal{L}(d)}\sum_{t\in\mathcal{T}_2} z_{d,\ell,t}$ and V the set of non-zero variables w associated with demand d.

Analysis. We first show that Algorithm 2 maintains feasibility of the solution for the LP relaxation (EC.3)-(EC.9). In particular, we will prove:

LEMMA EC.1. *For each iteration of Algorithm 2, if the iteration starts with a feasible solution for the LP relaxation* (EC.3)-(EC.9)*, then the solution at the end of the iteration is also a feasible solution for the LP relaxation* (EC.3)-(EC.9)*.*

Proof of Lemma EC.1. Given an iteration of Algorithm 2 that starts with a feasible solution for the LP relaxation, we will show that at the end of the iteration no constraint is violated. Let $\Delta_z = 1 - z^f_{d^*,\ell^*,t^*} \geq 0$ and $\Delta_w = 1 - w^f_{d^*,s^*} \geq 0$ denote the difference of these values from 1 at the beginning of the iteration. Since the algorithm rounds up at least one fractional variable among $z^f_{d^*,\ell^*,t^*}$ and $w^f_{d^*,s^*}$, this may lead to potential violation of the LP constraints. We will show that by decreasing appropriately other variables the Rounding Algorithm avoids such violations.

Constraints (EC.4) and (EC.5) for demand d^*: Note that for demand d^*, $z^f_{d^*,\ell^*,t^*}$ and $w^f_{d^*,s^*}$ both take value 1, and the algorithm rounds down all other variables to zero. As a result, these are the only non-zero variables associated with demand d^* at the end of the iteration, and therefore constraints (EC.4) and (EC.5) hold for demand d^*.

Constraints (EC.6): Assume at least one throughput constraint is violated due to rounding up $z^f_{d^*,\ell^*,t^*} = 1$. The only candidates for violation would be throughput constraints $(p, t^*) \in \mathcal{H}$ where $\ell^* \in p$, all of which would be hierarchical by Property 1. Let p' denote the smallest set for which the constraint is violated. Let $\eta_{p',t^*} < 1$ denote the absolute difference between the left-hand side and the right-hand side of the throughput constraint (p', t^*).

It is always feasible for the Rounding Algorithm to choose a subset of fractional variables z^f_{d,ℓ',t^*} with $\ell' \in p'$ whose sum is at least η_{p',t^*} so that we can decrease their total value by η_{p',t^*} to make sure constraint (p', t^*) is not violated. This can be proven by contradiction: Suppose not, i.e., all fractional variables z^f_{d,ℓ',t^*} for $\ell' \in p'$ sum up to $\alpha < \eta_{p',t^*}$. Then the total sum of the throughput in p' at time t^* is $\nu + \alpha$ for some $\nu \in \mathbb{N}$. By definition of η_{p',t^*}, $\nu + \alpha = \delta_{p',t^*,2} + \eta_{p',t^*}$. However, $\delta_{p',t^*,2}$ is an integer and $\eta_{p',t^*} < 1$, so $\alpha \geq \eta_{p',t^*}$, which is a contradiction. Therefore, the Rounding Algorithm must be able to find a subset of fractional variables z^f_{d,ℓ',t^*} with $\ell' \in p'$ that can be decreased to make the violated constraint tight and no longer violated.

Due to the hierarchical structure, decreasing the variables in p' also decreases the variables of all throughput constraints $(\tilde{p}, t^*) \in \mathcal{H}$ for which $\tilde{p} \supset p'$. Hence, the total decrease over all z^f variables in the largest of those sets would be at most Δ_z in order to ensure that all constraints are satisfied.

Constraints (EC.7): If the constraint for ℓ^* is violated, similar to the above, it is feasible for the Rounding Algorithm to find sufficient fractional variables to decrease and ensure that the constraint holds. The total decrease needs to be at most Δ_z since this is the maximum amount by which $\rho_{\ell^*,2}$ can be exceeded.

Constraints (EC.8): Finally, if the constraint for s^* does not hold, using same arguments as above, we know that there exist sufficient variables with fractional w^f_{d,s^*} that can be decreased for a total amount of at most Δ_w so that the constraint is satisfied.

Constraints (EC.4), (EC.5) for other demands: The variable decreases due to violated constraints (EC.6)-(EC.8) can lead to violation of constraints (EC.5) for the associated demands. Let d' denote one such demand for which constraint (EC.5) is violated. An important observation is that $d' \in D^f$, since only demands in D^f (with fractional z^f or w^f) can be decreased in our procedure.

By the hypothesis, at the start of the iteration constraint (EC.5) held with equality. Assume it is now violated due to a decrease on the z^f variables for demand d', i.e., $\sum_{\ell\in\mathcal{L}(d')}\sum_{t\in\mathcal{T}_2} z^f_{d',\ell,t} < \sum_{s\in\mathcal{S}(d')} w_{d',s}$. Algorithm 3 is invoked with $\kappa = \sum_{s\in\mathcal{S}(d')} w^f_{d',s} - \sum_{\ell\in\mathcal{L}(d')}\sum_{t\in\mathcal{T}_2} z^f_{d',\ell,t} > 0$ as the amount of violation and V the set of all non-zero variables (not necessarily fractional) w^f associated with demand d'. Since the sum of variables in V is $\sum_{s\in\mathcal{S}(d')} w^f_{d',s}$, which is strictly larger than κ, the algorithm can decrease the w^f variables sufficiently for the equality to hold. The case where constraint (EC.5) is violated due to decrease of the w^f variables can be dealt with similarly.

Using Lemma EC.1, we now show the following result:

Lemma EC.2. *Algorithm 2 terminates in a finite number of iterations with an integer solution that has objective value at least 1/4 of the optimal value of the LP relaxation* (EC.3)-(EC.9).

Proof of Lemma EC.2. We first show that the algorithm terminates in a finite number of iterations. Note that in each iteration the only demands whose values are affected (associated z^f or w^f variables increased or decreased) all belong to the set D^f. Moreover, the selected demand d^* at the end of the iteration has only integer z^f and w^f variables. As a result, the size of the set D^f decreases by at least one in each iteration and therefore the algorithm terminates with an integer solution in a finite number of steps. Specifically, the number of steps it requires is at most the number of demands with fractional z^f or w^f variables at the optimal solution of the LP relaxation that is given as input to the Rounding Algorithm.

Regarding the cost, we will compare the cost of the initial LP solution and the final integral solution of the Rounding Algorithm. Note that the variables whose values did not change during the execution of the algorithm contribute the same amount to the objective value of both the LP and the integral solution. We now discuss how the changes of the variable values affect the costs.

Consider an iteration of the Rounding Algorithm. Due to $z^f_{d^*,\ell^*,t^*}$ and $w^f_{d^*,s^*}$ taking value 1, we know that this demand contributes to the objective value of the final integral solution at least V_{d^*}.

In the LP solution, the same demand d^* contributed at most V_{d^*} to the objective, since $d^* \in D^f$ and V_{d^*} is the maximum value that can be obtained for this demand using the non-zero variables that contribute to the objective. We will now account for the contribution to the LP objective value of all other demands whose variable values were decreased during this iteration by the Rounding Algorithm. These can be categorized as follows:

Constraints (EC.6) and associated constraints (EC.5): We showed in Lemma EC.1 that the aggregate value by which the fractional values z^f_{d,ℓ,t^*} can be decreased due to violation of constraints (EC.6) in this iteration is at most Δ_z. This may trigger an associated total decrease of at most Δ_z to the w^f variables for the affected demands due to constraint (EC.5). Since all affected demands belong to D^f, by definition of V_{d^*}, the total value that the decreased values contributed to the LP objective is at most $\Delta_z \cdot V_{d^*}$.

Constraints (EC.7) and associated constraints (EC.5): Similarly, the total value of decreased $z^f_{d,\ell^*,t}$ is at most Δ_z, and the total value of decrease in the corresponding w^f variables is at most Δ_z. As a result, since all these variables correspond to $d \in D^f$, we have that the contribution to the LP objective of the reduced values before the change could have been at most $\Delta_z \cdot V_{d^*}$.

Constraints (EC.8) and associated constraints (EC.5): Finally, the total decrease in w^f_{d,s^*} variables is at most Δ_w, and similarly for the corresponding z^f variables. Since all these variables correspond to demands $d \in D^f$, we know that the contribution of the reduced values to the LP objective before the change could have been at most $\Delta_w \cdot V_{d^*}$.

Putting them all together, the actions of the Rounding Algorithm in each iteration contribute at least V_{d^*} to the final integral solution's cost for corresponding value of at most $V_{d^*} + \Delta_z \cdot V_{d^*} + \Delta_z \cdot V_{d^*} + \Delta_w \cdot V_{d^*} \leq 4 \cdot V_{d^*}$ of the LP objective value. Since this is true for all iterations, we conclude that the LP can have an optimal objective value at most 4 times that of the IP solution that we construct through the Rounding Algorithm, concluding the proof of Lemma EC.2.

With Lemma EC.2 at hand, the proof of Theorem EC.2 easily follows.

Proof of Theorem EC.2. Lemma EC.2 shows that given an optimal solution for the LP relaxation, we can construct an integral solution with at least 1/4 of the value. Since the value of the optimal IP solution is at least the value of the integral solution constructed by the Rounding Algorithm, we directly obtain that the integrality gap of the problem (EC.1)-(EC.2) is at most 4.