E-Companion to *"From Trees to Treewidth: Inventory Management in Complex Supply Chain Networks"*

Appendix A: Notation

Table EC.1 provides an overview of the notation used across the main paper.

Table EC.1 Notation used throughout the main paper.

	GSM (Section 2)
$G=(V,E)$, n	Acyclic directed graph with nodes V (with $n=\|V\|$) and arcs E.
$L\subseteq V$, d_j	Set of external customer-facing nodes with service time bounds d_j, $j\in L$.
T_j	Processing time at node $j\in V$.
D_j^t, μ_j, σ_j	Stochastic demand in period $t\in\mathbb{N}$ at node $j\in V$ (with mean μ_j and standard deviation σ_j).
$\bar{D}_j(\tau)$	Upper bound on demand at node $j\in V$ across τ periods.
B_j, $I_j(t)$	Base stock at node $j\in V$ and on-hand inventory at time $t\in\mathbb{N}$.
SI_j, S_j, M_j, Γ_j	Incoming and outgoing service times for node $j\in V$, with implied upper bounds.
M	Maximum replenishment time across nodes, with $M=\max_{j\in V} M_j$.
h_j, $f_j(SI_j-S_j)$	Per-unit inventory cost at node $j\in V$ and total cost of inventory as a function of service times.
	Tree case (Section 3)
$r_{k\ell}^j$	Indicator that $SI_j=\ell$ and $S_j=k$ for $j\in V$.
$q_{k\ell}^{ij}$	Indicator that $S_i=k$ and $SI_j=\ell$ for $(i,j)\in E$.
$\mathcal{S}_{ij}$	Set of indicators $q_{k\ell}^{ij}$ that must be zero due to $S_i\le SI_j$ for $(i,j)\in E$.
Δ_n, Δ	n-dimensional standard simplex.
	From trees to bounded treewidth (Sections 4.1–4.3)
G'	Intersection graph of G.
$\mathcal{T}=(T,\{X_z\}_{z\in T})$, $\omega=tw(G)$	Tree decomposition with tree T and a set X_z for each node $z\in T$, treewidth of G.
T_1, T_2	Partition of the tree T in a tree decomposition $\mathcal{T}$.
$s_{k\ell}^z$	Indicator for bag $z\in T$, equals $r_{k\ell}^j$ if $z\in T_1$ and $q_{k\ell}^{ij}$ if $z\in T_2$.
	General case (Section 4.4)
I_z, J_z	Indices of variables S_i (resp. SI_j) in bag X_z, for $z\in T$.
K_z, L_z, $\mathcal{D}_{K_z}$, $\mathcal{D}_{L_z}$	Vectors of values of variables in I_z (resp. J_z) and sets of possible values.
$s_{K_zL_z}^z$	Indicator that $S_{i_h}=k_{i_h}$ and $SI_{j_m}=\ell_{j_m}$ with $k_{i_h}\in K_z$ and $\ell_{j_m}\in L_z$ for $h=1,\dots,\|I_z\|$ and $m=1,\dots,\|J_z\|$.
$\mathcal{S}_{i_zj_z}^z$	Set of indicators $s_{K_zL_z}^z$ that must be zero due to $S_i\le SI_j$ for $(i,j)\in E$.

Appendix B: Preliminaries for the main results

In Section B.1, we provide definitions and show lemmas that will be useful for the proofs of Theorems 1 and 2. In Section B.2, we show Propositions 1 and 3, which appear in the main draft.

B.1. A few useful definitions and lemmas

DEFINITION EC.1. Let $u_0,u_1,\dots,u_n$ be $(n+1)$ affinely independent points. The simplex $\Delta_n(u_0,u_1,\dots,u_n)\subseteq\mathbb{R}^n$ is given by

$$\Delta_n(u_0,u_1,\dots,u_n)=\left\{\theta_0u_0+\theta_1u_1+\ldots+\theta_nu_n \;\middle|\; \sum_{i=0}^n\theta_i=1,\ \theta_i\ge 0,\forall i=0,\dots,n\right\}.$$

Note that if $u_0,\dots,u_n$ are integral, then $\Delta(u_0,\dots,u_n)$ is an integral polytope as it is simply the convex hull of $u_0,\dots,u_n$. The *standard* simplex $\Delta_n\subseteq\mathbb{R}^{n+1}$, as introduced in Section 3, refers to the setting where $u_0,\dots,u_n$ are the $(n+1)$ standard unit vectors in $\mathbb{R}^{n+1}$.

LEMMA EC.1. *Let $\Delta_{n-1} \subseteq \mathbb{R}^n$ be the standard simplex and let $V \in \{0,1\}^{m\times n}$ be a binary matrix. Define*

$$Q := \{(x,y) \in \mathbb{R}^{n+m} \mid x \in \Delta_{n-1}, y = Vx\}.$$

Then, $Q \subseteq \mathbb{R}^{n+m}$ is an $(n-1)$-dimensional simplex with binary vertices $\begin{bmatrix} e_i \\ v_i \end{bmatrix}$ for $i = 1,\dots,n$, where e_i is the i^{th} standard unit vector in $\mathbb{R}^n$ and v_i is the i^{th} column of V.

This result implies that Q thus defined is integral.

Proof. Let $u_i = \begin{bmatrix} e_i \\ Ve_i \end{bmatrix} = \begin{bmatrix} e_i \\ v_i \end{bmatrix}$ for $i = 1,\dots,n$. Note that $u_i \in \{0,1\}^{n+m}$ as $V \in \{0,1\}^{m\times n}$. Furthermore, $u_1,\dots,u_n$ are affinely independent due to the fact that $e_1,\dots,e_n$ are affinely independent. We now show that $Q = \Delta(u_1,\dots,u_n)$. If $(x,y) \in Q$, then there exist $\theta_1,\dots,\theta_n$ such that $\sum_{i=1}^n \theta_i = 1,\ \theta_i \geq 0,\ \forall i = 1,\dots,n$, and $x = \sum_{i=1}^n \theta_i e_i$ as $x \in \Delta_{n-1}$. We then have $y = Vx = \sum_{i=1}^n \theta_i Ve_i$, which implies that $(x,y) = \sum_{i=1}^n \theta_i u_i$ and $(x,y) \in \Delta(u_1,\dots,u_n)$. Conversely, if $(x,y) \in \Delta(u_1,\dots,u_n)$, then there exist $\theta_1,\dots,\theta_n$ such that $\sum_{i=1}^n \theta_i = 1, \theta_i \geq 0,\ \forall i = 1,\dots,n$, and $(x,y) = \sum_{i=1}^n \theta_i u_i$. It follows from this that $x = \sum_{i=1}^n \theta_i e_i$, i.e., $x \in \Delta_{n-1}$ and $y = \sum_{i=1}^n \theta_i Ve_i = Vx$. This in turn implies that $(x,y) \in Q$. □

LEMMA EC.2. *Let $u_0,u_1,\dots,u_k \in \{0,1\}^n$ be affinely independent vectors for $k \leq n$. Let $P = \Delta_k(u_0,\dots,u_k)$ and let $S \subseteq \{1,\dots,n\}$. Then,*

$$Q := \{x \in \mathbb{R}^n \mid x \in P, x_i = 0, \forall i \in S\}$$

is a simplex whose vertices are a subset of $u_0,u_1,\dots,u_k \in \{0,1\}^n$.

Proof. Let $S_1 = \{k' \in \{0,\dots,k\} \mid u^i_{k'} = 0, \forall i \in S\}$. In other words, S_1 contains the indexes of a subset of the vertices of P for which all i^{th} entries are zero for any $i \in S$. As $\{u_{k'}\}_{k'\in S_1}$ is a subset of $u_0,\dots,u_k$, this set remains affinely independent. Letting $Q' = \Delta_{|S_1|-1}(\{u_{k'}\}_{k'\in S_1})$, we show that $Q' = Q$. This proves that Q is a simplex. It is straightforward to see that $Q' \subseteq Q$. For $Q \subseteq Q'$, let $x \in Q$: we have $x = \sum_{k'=0}^k \theta_{k'} u_{k'}$ where $\sum_{k'=0}^k \theta_{k'} = 1$ and $\theta_{k'} \geq 0, \forall k' = 0,\dots,k$. Let $k_0 \notin S_1$. By definition, there exists $i_{k_0} \in S$ such that $u^{i_{k_0}}_{k_0} \neq 0$, i.e., $u^{i_{k_0}}_{k_0} = 1$ as $u_0,\dots,u_k$ are binary. Thus,

$$x_{i_{k_0}} = \sum_{k'\in S_1} \theta_{k'} u^{i_{k_0}}_{k'} + \sum_{k'\notin S_1} \theta_{k'} u^{i_{k_0}}_{k'} = \sum_{k'\notin S_1} \theta_{k'} u^{i_{k_0}}_{k'} = \theta_{k_0} + \sum_{k'\notin S_1, k'\neq k_0} \theta_{k'} u^{i_{k_0}}_{k'} = 0.$$

As $\theta_{k_0} \geq 0$ and $\sum_{k'\notin S_1, k'\neq k_0} \theta_{k'} u^{i_{k_0}}_{k'} \geq 0$, it must be the case that $\theta_{k_0} = 0$. We repeat this procedure for any $k' \notin S_1$. It follows that $\theta_{k'} = 0, \forall k' \notin S_1$ and thus $x \in Q'$. □

LEMMA EC.3. *Let $u_0,u_1,\dots,u_k \in \{0,1\}^n$ be affinely independent vectors for $k \leq n$. Let $P = \Delta_k(u_0,\dots,u_k)$ and let $S' \subseteq \{1,\dots,n\}$. Further, let $b_i \in \{0,1,\dots\}$ for any $i \in S'$. Then, $Q' := \{x \in \mathbb{R}^n \mid x \in P, x_i \leq b_i, \forall i \in S'\}$ is a simplex whose vertices are a subset of $u_0,u_1,\dots,u_k \in \{0,1\}^n$.*

Proof. This is a straightforward corollary of Lemma EC.2 by choosing $S = \{i \in S' \mid b_i = 0\}$. Then, note that $Q' = \{x \in \mathbb{R}^n \mid x \in P, x_i = 0, \forall i \in S\}$. This is due to the fact that the vertices of P are binary, so we have that $0 \leq x_i \leq 1$ for any $i = 1,\dots,n$ from $x \in P$. Thus, if $b_i \geq 1$, $x_i \leq b_i$ is a redundant constraint and if $b_i = 0$, then $x_i = 0$. □

Recall the definition of the projection of a polytope $Q \subseteq \mathbb{R}^{n+m}$ onto the variables $x \in \mathbb{R}^n$:

$$proj_x(Q) = \left\{ x \in \mathbb{R}^n \mid \exists y \in \mathbb{R}^m \text{ s.t. } \begin{pmatrix} x \\ y \end{pmatrix} \in Q \right\}.$$

LEMMA EC.4. *Let $P \subseteq \mathbb{R}^n$ be an integral polytope, and let $Q \subseteq \mathbb{R}^{m+n}$ be an integral polytope. If for any integral $x \in P$, $\exists$ an integral y such that $\begin{pmatrix} x \\ y \end{pmatrix} \in Q$, and conversely, if for any integral $\begin{pmatrix} x \\ y \end{pmatrix} \in Q$, $x \in P$, then $P = proj_x(Q)$.*

Proof. Let $x_0 \in P$. As P is an integral polytope, there exist integral vertices $v_1, \dots, v_N$ of P such that $x_0 = \sum_{i=1}^N \lambda_i v_i$, where $\sum_{i=1}^N \lambda_i = 1$ and $\lambda_i \geq 0$, $\forall i = 1, \dots, N$. As v_i is integral and in P, there exists y_i integral such that $\begin{pmatrix} v_i \\ y_i \end{pmatrix} \in Q$. As Q is convex, $\sum_{i=1}^N \lambda_i \begin{pmatrix} v_i \\ y_i \end{pmatrix} = \begin{pmatrix} x_0 \\ \sum_{i=1}^N \lambda_i y_i \end{pmatrix} \in Q$. Thus $x_0 \in proj_x(Q)$. Now, let $x_0 \in proj_x(Q)$, i.e., there exists y such that $\begin{pmatrix} x_0 \\ y \end{pmatrix} \in Q$. As Q is an integral polytope, we can write $\begin{pmatrix} x_0 \\ y \end{pmatrix} = \sum_{i=1}^M \gamma_i \begin{pmatrix} u_i \\ w_i \end{pmatrix}$ where $\begin{pmatrix} u_i \\ w_i \end{pmatrix}$ are integral and in Q, and $\sum_{i=1}^M \gamma_i = 1, \gamma_i \geq 0, \forall i = 1, \dots, M$. Thus $u_i \in P$ and, as P is convex, $x_0 = \sum_{i=1}^M \gamma_i u_i \in P$. □

The final lemma requires us to define the notion of a *join* of two polytopes.

DEFINITION EC.2. Let $Q_1 \subseteq \mathbb{R}^{n_1} \times \mathbb{R}^p$ and $Q_2 \subseteq \mathbb{R}^{n_2} \times \mathbb{R}^p$ be two polytopes on variable sets (x_1, y) and (x_2, y), respectively and suppose that Q_1 and Q_2 both project onto the same set in the y-space, then

$$Q_1 \wedge Q_2 := \{(x_1, x_2, y) \mid (x_1, y) \in Q_1, (x_2, y) \in Q_2\}.$$

LEMMA EC.5 **(Lemma APDX.5 in Kim et al. 2025).** *For $i = 1, 2$, let Q_i be an integral polytope in variables $(x_i, y) \in \mathbb{R}^{p_i} \times \mathbb{R}^{p_0}$. Assume that $proj_y(Q_1) = proj_y(Q_2) = \Delta$, where Δ is some simplex. Then the join $Q_1 \wedge Q_2$ is integral.*

B.2. Proofs of Propositions 1 and 3

Proof of Proposition 1. The first step ensures that each original decision variable S_i or SI_i appears as a node in G'. The first step also captures the constraints ($2a_j$) and the separable objective, since it creates edges between (S_j, SI_j) in G' for each $j = 1, \dots, n$. The second step captures constraints ($2b_{ij}$) since we add edge (S_i, SI_j) to G' for each $(i, j) \in E$. Constraints ($2c_j$) and ($2d_j$) have only one decision variable and do not create edges in the intersection graph. Hence, G' matches the definition of the intersection graph in Definition 2. □

Proof of Proposition 3. It is easy to see that $\mathcal{T}$ has width $2\omega + 1$ as each bag in $\mathcal{T}^*$ contains at most $\omega + 1$ variables. Thus, each bag in $\mathcal{T}$ contains at most $2\omega + 2$ variables, and hence has width at most $2\omega + 1$.

To see that $\mathcal{T}$ is a tree decomposition of G', first note that as $\cup_{z \in T^*} X_z^* = \{1, \dots, n\}$, we have $\cup_{z \in T} X_z = \{S_1, SI_1, \dots, S_n, SI_n\}$. Furthermore, two types of edges appear in G': edges between SI_i and S_i for $i \in V$ and edges between S_i and SI_j for $(i, j) \in E$. As any $i \in V$ appears in a bag X_z^* for some $z \in T^*$, and we replace i by S_i and SI_i, it follows that S_i and SI_i appear in the same bag at least once. Likewise, the pair $(i, j) \in E$ must appear in the same bag X_z^* for some z as $\mathcal{T}^*$ is a tree decomposition of G. The replacement of i and j by S_i, SI_i, S_j and SI_j ensures that there is an edge between S_i and SI_j for any $(i, j) \in E$. It remains to argue that if S_i (resp. SI_i) appears in two bags X_z and $X_{z'}$ for $z, z' \in T$, there exists a path between z and z' such that S_i (resp. SI_i) appears in each bag on that path. To see this, suppose that there exists $\hat{z}$ on the path between z and z' such that S_i does not appear in $X_{\hat{z}}$. As S_i appears in both X_z and $X_{z'}$, but not in $X_{\hat{z}}$, it means that i appears in both X_z^* and $X_{z'}^*$, but not in $X_{\hat{z}}^*$. This violates the fact that $\mathcal{T}^*$ is a tree decomposition of G. □

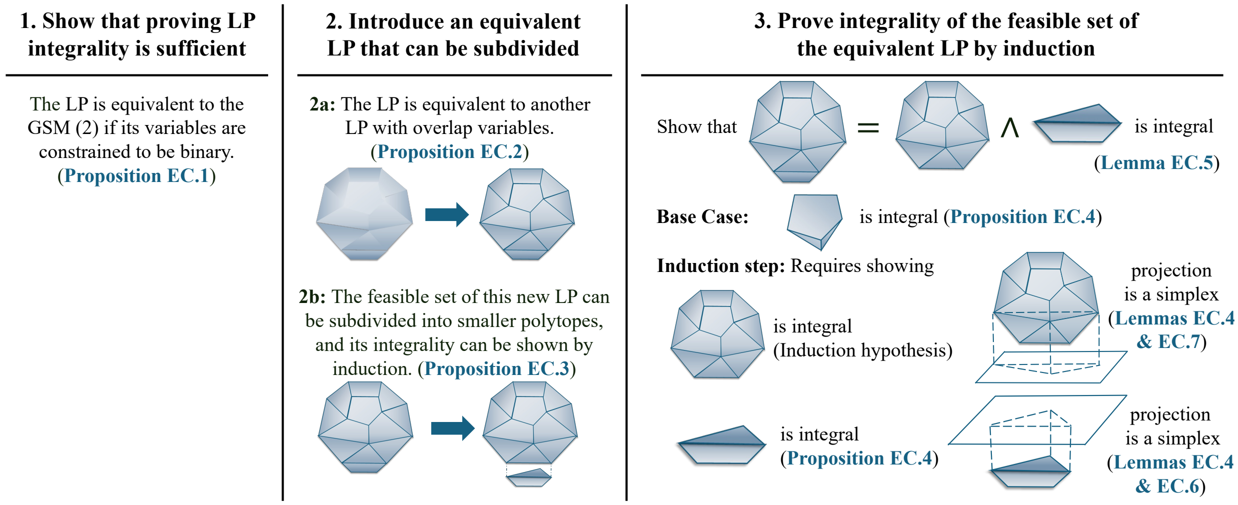
Appendix C: Proof of Theorem 1

We present the proof of Theorem 1, whose structure mirrors the overview in Figure EC.1.

Showing Step 1

PROPOSITION EC.1. *Problems (4) and (2) are equivalent, provided that the variables appearing in (4) are constrained to be integers.*

Figure EC.1 Structure of the proof of Theorem 1, annotated with the relevant propositions and lemmas



Proof. Let S_j, SI_j be a solution to (2) for $j \in V$. We let:

$$r^j_{k\ell} = \mathbf{1}_{S_j=k, SI_j=\ell}, \ \forall k, \ell, j$$
$$q^{ij}_{k\ell} = \mathbf{1}_{S_i=k, SI_j=\ell}, \ \forall k, \ell, (i,j) \in E, \tag{EC.1}$$

where $\mathbf{1}$ is the indicator function. It is clear that ($4c_j$) holds for $j \in V$ such that $(i,j) \in E$, as $\sum_{k=0}^{\Gamma_j} r^j_{k\ell} = 1_{SI_j=\ell} = \sum_{k=0}^{\Gamma_i} q^{ij}_{k\ell}, \forall \ell$. Likewise, ($4d_i$) holds for $i \in V$ such that $(i,j) \in E$, as $\sum_{\ell=0}^{M_i} r^i_{k\ell} = 1_{S_i=k} = \sum_{\ell=0}^{M_j} q^{ij}_{k\ell}, \forall \ell$. Furthermore, ($4e_j$)-($4f_{ij}$) hold by construction and they are integer variables. Now, let $(i,j) \in E$ and suppose that ($2b_{ij}$) holds. Moreover, suppose that $SI_j = \ell$ and $S_i = k$ for some k, ℓ. As $SI_j - S_i \geq 0$, we must have that $\ell \geq k$. Thus, it cannot be that $k > \ell$ and so $q^{ij}_{k\ell} = 0$ when $k > \ell$. This is required by ($4b_{ij}$) via the definition of $\mathcal{S}_{ij}$ in (3). Now, assume that constraint ($2a_j$) holds for $j \in V$ and suppose that $SI_j = \ell$ and $S_j = k$ for some k, ℓ. Constraint ($2a_j$) is trivially satisfied if $k \leq \ell$. We, thus, assume moving forward that $k > \ell$. We have, from constraint ($2a_j$), that $k - \ell \leq T_j$. Thus, it cannot be that $k - \ell > T_j \geq 0$. This implies that $r^j_{k\ell} = 0$ when $T_j/(k-\ell) < 1$, which in turn implies that $r^j_{k\ell} \leq \lfloor T_j/(k-\ell) \rfloor$, i.e., we get ($4a_j$). Finally, it is straightforward to see that the objective values of (2) and (4) are equal, by noting that

$$\sum_{\ell=0}^{M_j} \sum_{k=0}^{\Gamma_j} f_j(\ell - k) r^j_{kl} = \sum_{\ell=0}^{M_j} \sum_{k=0}^{\Gamma_j} f_j(\ell - k) 1_{S_j=k, SI_j=\ell} = f_j(SI_j - S_j), \forall j \in V.$$

Now let $\{r^j_{k\ell}\}, \{q^{ij}_{k\ell}\}$ be an *integer* solution to (4). The constraints ($4e_j$) and ($4f_{ij}$) together with integrality imply that these variables are in fact binary. Define SI_j, S_j for $j \in V$ as in (5). Constraints ($2c_j$) and ($2d_j$) follow immediately. For constraint ($2b_{ij}$), let $(i,j) \in E$. From ($4c_j$) and ($4d_i$), we have that:

$$SI_j - S_i = \sum_{\ell=0}^{M_j} \ell \cdot \left(\sum_{k=0}^{\Gamma_i} q^{ij}_{k\ell} \right) - \sum_{k=0}^{\Gamma_i} k \cdot \left(\sum_{\ell=0}^{M_j} q^{ij}_{k\ell} \right) = \sum_{\ell=0}^{M_j} \sum_{k=0}^{\Gamma_i} (\ell - k) q^{ij}_{k\ell}.$$

From the definition of $\mathcal{S}_{ij}$, constraint ($4b_{ij}$) implies that, if $k > \ell$, $q^{ij}_{k\ell} = 0$. Thus, $SI_j - S_i \geq 0$. For constraint ($2a_j$), let $j \in V$. Constraints ($4e_j$), together with the fact that $r^j_{k\ell}$ are binary, imply that there exist $\ell_j \in \{0, \ldots, M_j\}$ and $k_j \in \{0, \ldots, \Gamma_j\}$ such that $r^j_{k_j \ell_j} = 1$ with $r^j_{k\ell} = 0$ for any $(k, \ell) \neq (k_j, \ell_j)$. For S_j, SI_j as defined, this implies that $S_j = k_j$ and $SI_j = \ell_j$. If $k_j \leq \ell_j$, then $S_j - SI_j = k_j - \ell_j \leq T_j$ trivially holds as $T_j \geq 0$. Now, suppose that $k_j > \ell_j$. Constraint ($4a_j$) implies that $r^j_{k_j \ell_j} \leq T_j/(k_j - \ell_j)$, i.e., $(k_j - \ell_j) = S_j - SI_j \leq T_j$, so ($2a_j$). Finally, the objective value of both (2) and (4) are the same, because $r^j_{k\ell} = 1$ if and only if $S_j = k$ and $SI_j = \ell$, following the same reasoning as above. □

Showing Step 2a

We introduce a linear program, which is similar to (4), except that it has additional variables $\{\lambda_\ell^j\}$ and $\{\gamma_k^j\}$ acting as indicator variables of SI_j and S_j:

$$\min_{\{\lambda_\ell^j\},\{\gamma_k^j\},\{r_{k\ell}^j\},\{q_{k\ell}^{ij}\}} \quad \sum_{j\in V}\sum_{\ell=0}^{M_j}\sum_{k=0}^{\Gamma_j} f_j(\ell-k) r_{k\ell}^j \tag{EC.2}$$

$$\text{s.t.} \quad r_{k\ell}^j \le \left\lfloor \frac{T_j}{k-\ell} \right\rfloor,\ \forall k=0,\dots,\Gamma_j,\ \ell=0,\dots,M_j,\ k>\ell,\ j\in V, \tag{$EC.2a_j$}$$

$$q_{k\ell}^{ij}=0,\ \forall (k,\ell)\in \mathcal{S}_{ij},\ \forall (i,j)\in E, \tag{$EC.2b_{ij}$}$$

$$\lambda_\ell^j = \sum_{k=0}^{\Gamma_i} q_{k\ell}^{ij},\ \forall \ell=0,\dots,M_j,\ \forall j\in V \text{ s.t. } (i,j)\in E, \tag{$EC.2c_j$}$$

$$\lambda_\ell^j = \sum_{k=0}^{\Gamma_j} r_{k\ell}^j,\ \forall \ell=0,\dots,M_j,\ \forall j\in V, \tag{$EC.2d_j$}$$

$$\gamma_k^i = \sum_{\ell=0}^{M_j} q_{k\ell}^{ij},\ \forall k=0,\dots,\Gamma_i,\ \forall i\in V \text{ s.t. } (i,j)\in E, \tag{$EC.2e_i$}$$

$$\gamma_k^j = \sum_{\ell=0}^{M_j} r_{k\ell}^j,\ \forall k=0,\dots,\Gamma_j,\ \forall j\in V, \tag{$EC.2f_j$}$$

$$\{r_{k\ell}^j\}_{k,\ell}\in\Delta,\ \forall j\in V, \tag{$EC.2g_j$}$$

$$\{q_{k\ell}^{ij}\}_{k,\ell}\in\Delta,\ \forall (i,j)\in E. \tag{$EC.2h_{ij}$}$$

We denote by P_{lin} its feasible set and show that (EC.2) is equivalent to (4).

PROPOSITION EC.2. *The LP (EC.2) is equivalent to (4). Furthermore, if the feasible set of (EC.2) is integral then the feasible set of (4) is integral.*

Proof. Let Q be the feasible set of (4). Note that

$$P_{lin} = \left\{ \{\lambda_\ell^j\},\{\gamma_k^j\},\{r_{k\ell}^j\},\{q_{k\ell}^{ij}\} \mid \{r_{k\ell}^j\},\{q_{k\ell}^{ij}\}\in Q,\ \lambda_\ell^j = \sum_{k=0}^{\Gamma_i} q_{k\ell}^{ij}\forall \ell,j,\ \gamma_k^i = \sum_{\ell=0}^{M_j} r_{k\ell}^j, \forall k,j \right\}.$$

It is thus straightforward to see that any feasible point of (4) maps to a feasible point of (EC.2), and vice-versa. As the objective functions of (4) and (EC.2) are equal, the equivalence follows.

For the second statement, if $\{\{r_{k\ell}^j\},\{q_{k\ell}^{ij}\}\}$ is a vertex of Q, then $\{\{r_{k\ell}^j\},\{q_{k\ell}^{ij}\},\{\sum_{k=0}^{\Gamma_i} q_{k\ell}^{ij}\},\{\sum_{\ell=0}^{M_j} r_{k\ell}^j\}\}$ is a vertex of P_{lin}. This can be seen, e.g., through a proof by contradiction. In the same way, the converse can be shown: if $\left\{\{\lambda_\ell^j\},\{\gamma_k^j\},\{r_{k\ell}^j\},\{q_{k\ell}^{ij}\}\right\}$ is an integral vertex of P_{lin}, then $\left\{\{r_{k\ell}^j\},\{q_{k\ell}^{ij}\}\right\}$ is an integral vertex of Q. Thus, if P_{lin} is integral, that is, each one of its vertices is integer-valued, then Q is integral: any integer vertex of P_{lin} maps to an integer vertex of Q, and all vertices of P_{lin} correspond to vertices of Q. □

Showing Step 2b

Recalling that P_{lin} is the feasible set of (EC.2), we define the polytopes

$$Q_j^1 = \left\{ \{\lambda_\ell^j\}_\ell,\{\gamma_k^j\}_k,\{r_{k\ell}^j\}_{k,\ell} \mid (EC.2a_j),(EC.2d_j),(EC.2f_j),(EC.2g_j) \right\} \text{ for } j\in V,$$

$$Q_{ij}^2 = \left\{ \{\lambda_\ell^j\}_\ell,\{\gamma_k^i\},\{q_{k\ell}^{ij}\}_{k,\ell} \mid (EC.2b_{ij}),(EC.2c_j),(EC.2e_i),(EC.2h_{ij}) \right\} \text{ for } (i,j)\in E.$$

We show that P_{lin} can be written as the *join* (see Definition EC.2) of these polytopes.

PROPOSITION EC.3. *We have that* $P_{lin} = \left(\wedge_{j\in V} Q^1_j\right) \wedge \left(\wedge_{(i,j)\in E} Q^2_{ij}\right)$.

Proof. This is immediate by noting that any variable appearing in P_{lin} appears in at least one of Q^1_j or Q^2_{ij} (or both). All constraints apply as they can also be associated to either edges or nodes of the graph. This shows that $P_{lin} \subseteq \left(\wedge_{j\in V} Q^1_j\right) \wedge \left(\wedge_{(i,j)\in E} Q^2_{ij}\right)$. For the converse, note that all variables and constraints that appear in Q^1_j and Q^2_{ij} also feature in P_{lin}. □

Showing Step 3

The proof of integrality of P_{lin} requires a few intermediary propositions (see Figure EC.1), which we prove now. Our first proposition is useful for both the base and induction steps.

PROPOSITION EC.4. *We have that:*

(i) Q^1_j *is integral for any* $j \in V$.

(ii) Q^2_{ij} *is integral for any* $(i,j) \in E$.

Proof. We show (i) first. Let $j \in V$. Without the constraints ($EC.2a_j$), the polytope is a simplex with binary vertices (Lemma EC.1). Adding back the constraints maintains the fact that the polytope is a simplex with binary vertices (Lemma EC.3), which implies that Q^1_j is integral. For (ii), let $(i,j) \in E$. If we drop the constraints ($EC.2b_{ij}$), we obtain a polytope with binary vertices (Lemma EC.1). Adding back these constraints maintains the fact that Q^2_{ij} is a simplex with binary vertices (Lemma EC.2). Thus Q^2_{ij} is integral. □

The next two lemmas are used for the induction step. They require an ordering of the nodes of G that ensures that the labels of the parents are before those of the children. This can be obtained, e.g., by taking a level-order transversal. Given a label $p \in \{1,\dots,n\}$, we let G_p denote the subtree of G consisting of nodes 1 through p. The first lemma shows that we can extend different partial integral solutions to integral points in Q^1_j and Q^2_{ij}.

LEMMA EC.6. *For fixed* $j \in V$ *and* $(i,j) \in E$*, we have the following:*

(i) If $\{\lambda^j_\ell\}_\ell \in \Delta$ *is integral, then there exist integral* $\{\gamma^j_k\}_k, \{r^j_{k\ell}\}_{k,\ell}$ *such that* $\left\{\{\lambda^j_\ell\}_\ell, \{\gamma^j_k\}_k, \{r^j_{k\ell}\}_{k,\ell}\right\} \in Q^1_j$.

(ii) If $\{\gamma^j_k\}_k \in \Delta$ *is integral, then there exist integral* $\{\lambda^j_\ell\}_\ell, \{r^j_{k\ell}\}_{k,\ell}$ *such that* $\left\{\{\lambda^j_\ell\}_\ell, \{\gamma^j_k\}_k, \{r^j_{k\ell}\}_{k,\ell}\right\} \in Q^1_j$.

(iii) If $\{\lambda^j_\ell\}_\ell \in \Delta$ *is integral, then there exist integral* $\{\gamma^i_k\}_k, \{q^{ij}_{k\ell}\}_{k,\ell}$ *such that* $\left\{\{\lambda^j_\ell\}_\ell, \{\gamma^i_k\}_k, \{q^{ij}_{k\ell}\}_{k,\ell}\right\} \in Q^2_{ij}$.

(iv) If $\{\gamma^i_k\}_k \in \Delta$ *is integral, then there exist integral* $\{\lambda^j_\ell\}_\ell, \{q^{ij}_{k\ell}\}_{k,\ell}$ *such that* $\left\{\{\lambda^j_\ell\}_\ell, \{\gamma^i_k\}_k, \{q^{ij}_{k\ell}\}_{k,\ell}\right\} \in Q^2_{ij}$.

Proof. For (i), if $\{\lambda^j_\ell\}_\ell \in \Delta$ is integral, then there exists ℓ_j such that $\lambda^j_{\ell_j} = 1$ and $\lambda^j_\ell = 0$ for all $\ell \neq \ell_j$. Take $r^j_{0\ell_j} = 1$ and $r^j_{k\ell} = 0$ for all $(k,\ell) \neq (0,\ell_j)$, thus $\gamma^j_0 = 1$ and $\gamma^j_k = 0$ for any $k \neq 0$. Note that these values are integers and it is straightforward to check that $\left\{\{\lambda^j_\ell\}_\ell, \{\gamma^j_k\}_k, \{r^j_{k\ell}\}_{k,\ell}\right\} \in Q^1_j$.

For (ii), if $\{\gamma^j_k\}_k \in \Delta$ is integral, then there exists k_j such that $\gamma^j_{k_j} = 1$ and $\gamma^j_k = 0$ for all $k \neq k_j$. Take $r^j_{k_j M_j} = 1$ and $r^j_{k\ell} = 0$ for all $(k,\ell) \neq (k_j, M_j)$, thus $\lambda^j_{M_j} = 1$ and $\lambda^j_\ell = 0$ for any $\ell \neq M_j$. Note that these values are integers and it is straightforward to check that $\left\{\{\lambda^j_\ell\}_\ell, \{\gamma^j_k\}_k, \{r^j_{k\ell}\}_{k,\ell}\right\} \in Q^1_j$.

For (iii), if $\{\lambda^j_\ell\}_\ell \in \Delta$ is integral, then there exists ℓ_j such that $\lambda^j_{\ell_j} = 1$ and $\lambda^j_\ell = 0$ for all $\ell \neq \ell_j$. Take $q^{ij}_{0\ell_j} = 1$ and $q^{ij}_{k\ell} = 0$ for all $(k,\ell) \neq (0,\ell_j)$, thus $\gamma^i_0 = 1$ and $\gamma^i_k = 0$ for any $k \neq 0$. Note that these values are integers and it is straightforward to check that $\left\{\{\lambda^j_\ell\}_\ell, \{\gamma^i_k\}_k, \{q^{ij}_{k\ell}\}_{k,\ell}\right\} \in Q^2_{ij}$.

For (iv), if $\{\gamma^i_k\}_k \in \Delta$ is integral, then there exists k_j such that $\gamma^{i_j}_{k_j} = 1$ and $\gamma^i_k = 0$ for all $k \neq k_j$. Take $q^{ij}_{k_j M_j} = 1$ and $q^{ij}_{k\ell} = 0$ for all $(k,\ell) \neq (k_j, M_j)$, thus $\lambda^j_{M_j} = 1$ and $\lambda^j_\ell = 0$ for any $\ell \neq M_j$. Note that these values are integers and it is straightforward to check that $\left\{\{\lambda^j_\ell\}_\ell, \{\gamma^i_k\}_k, \{q^{ij}_{k\ell}\}_{k,\ell}\right\} \in Q^2_{ij}$. □

The next lemma extends this result to the subtree G_p:

LEMMA EC.7. *Let $p \in \{1,\dots,n\}$ and let G_p be the subtree of G containing nodes $\{1,\dots,p\}$. Let i_p be the parent of p in G_p. We have that:*

(i) If $\{\lambda^p_\ell\}_\ell \in \Delta$ is integral, then there exist $\left\{\{\lambda^j_\ell\}_{\ell, j\neq p \in G_p}, \{\gamma^i_k\}_{k, i\in G_p}, \{q^{ij}_{k\ell}\}_{k,\ell,(i,j)\in G_p}, \{r^j_{k\ell}\}_{k,\ell,j\in G_p}\right\} \in (\wedge_{j\in G_p} Q^1_j) \wedge (\wedge_{(i,j)\in G_p} Q^2_{ij})$ that are integral.

(ii) If $\{\gamma^{i_p}_k\}_\ell \in \Delta$ is integral, then there exist $\left\{\{\lambda^j_\ell\}_{\ell, j\in G_{p-1}}, \{\gamma^i_k\}_{k, i\in G_{p-1}}, \{q^{ij}_{k\ell}\}_{k,\ell,(i,j)\in G_p}, \{r^j_{k\ell}\}_{k,\ell,j\in G_{p-1}}\right\} \in \left((\wedge_{j\in G_{p-1}} Q^1_j) \wedge (\wedge_{(i,j)\in G_{p-1}} Q^2_{ij})\right) \wedge Q^2_{i_p p}$ that are integral.

Proof. We first suppose that $\{\lambda^p_\ell\}_\ell \in \Delta$ is integral. We then proceed by induction on p. When $p = 1$, G_p only has one node. The result then follows from Lemma EC.6 (i). Now, let

$$U_p = (\wedge_{j\in G_p} Q^1_j) \wedge (\wedge_{(i,j)\in G_p} Q^2_{ij}).$$

Suppose the result holds for $p-1$ and we show that it holds for p. Note that $U_p = U_{p-1} \wedge Q^1_p \wedge Q^2_{i_p p}$. From Lemma EC.6 (i), there exist integral $\{\gamma^p_k\}_k, \{r^p_{k\ell}\}_{k,\ell}$ such that $\left\{\{\lambda^p_\ell\}_\ell, \{\gamma^p_k\}_k, \{r^p_{k\ell}\}_{k,\ell}\right\} \in Q^1_p$. Observe that $\{\lambda^p_\ell\}_\ell$ is shared across Q^1_p and $Q^2_{i_p p}$ and that it is in Δ. From Lemma EC.6 (iii), there exist $\{\gamma^{i_p}_k\}_k, \{q^{i_p p}_{k\ell}\}_{k,\ell}$ such that $\left\{\{\lambda^{i_p}_\ell\}_\ell, \{\gamma^{i_p}_k\}_k, \{q^{i_p p}_{k\ell}\}_{k,\ell}\right\} \in Q^2_{i_p p}$. Combining this with the induction assumption, we obtain our conclusion.

Now suppose that $\{\gamma^{i_p}_k\} \in \Delta$ is integral. We use induction once again to show that the result follows, letting

$$V_p = (\wedge_{j\in G_{p-1}} Q^1_j) \wedge (\wedge_{(i,j)\in G_{p-1}} Q^2_{ij}) \wedge Q^2_{i_p p}.$$

For $p = 2$, we have $V_2 = Q^1_1 \wedge Q^2_{12}$. The result holds straightforwardly from a combination of Lemma EC.6 (iv) and (ii), noting that the overlap between the two sets is $\{\gamma^1_k\}_k$. For general p, the result follows from similar arguments to above, using Lemma EC.6 (iv) and (ii). □

We now have all the elements to show Theorem 1:

Proof of Theorem 1. It is enough to show that the feasible set P_{lin} of (EC.2) is integral. Indeed, Proposition EC.2 then implies that the feasible set of (4) is integral. From Proposition EC.1, it follows that solving (4) is equivalent to showing (2), which gives us Theorem 1. To show that P_{lin} is integral, consider once again the subtree G_p of G and the parent i_p of p in G. Define for $p \in \{1,\dots,n\}$

$$U_p := \left(\wedge_{\{j\in G_p\}} Q^1_j\right) \wedge \left(\wedge_{\{(i,j)\in G_p\}} Q^2_{ij}\right).$$

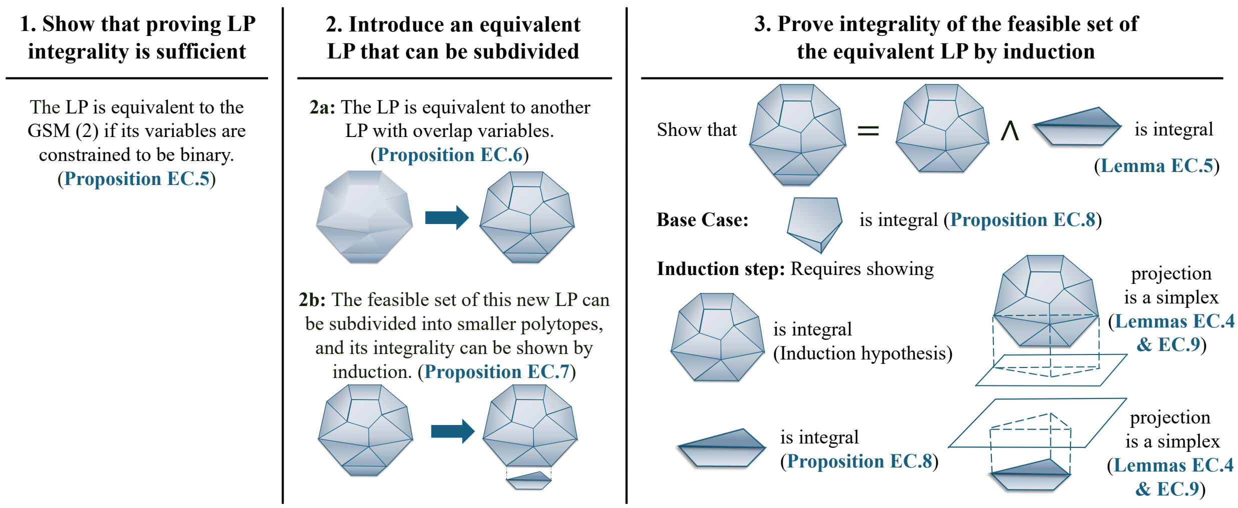
From Proposition EC.3, it is clear that $U_n = P_{lin}$. We now show by induction that U_n is integral.

Base Case: Suppose that $p = 1$. Then G_1 is simply node 1 and $U_1 = Q^1_1$, which we know to be integral from Proposition EC.4.

Induction Step: Let $p \in \{1,\dots,n\}$ and suppose that U_{p-1} is integral. We show that

$$U_p = U_{p-1} \wedge Q^1_p \wedge Q^2_{i_p p}$$

Figure EC.2 Structure of the proof of Theorem 2, annotated with the relevant propositions and lemmas



is integral too. We use Lemma EC.5 to show successively that $U_{p-1} \wedge Q^2_{i_p p}$ is integral, and then that $(U_{p-1} \wedge Q^2_{i_p p}) \wedge Q^1_p$ is integral. We start with $U_{p-1} \wedge Q^2_{i_p p}$. Note that the common variables between U_{p-1} and $Q^2_{i_p p}$ are $\{\gamma_k^{i_p}\}_k$. Further note that $Q^2_{i_p p}$ is integral from Proposition EC.4 and that U_{p-1} is integral by assumption. It is clear that $proj_{\{\gamma_k^{i_p}\}} Q^2_{i_p p} = \Delta$. This follows from Lemma EC.4 and Lemma EC.6 (iv). Likewise, $proj_{\{\gamma_k^{i_p}\}} U_{p-1} = \Delta$. This follows from Lemma EC.4 once again and Lemma EC.7 (ii). The conclusion that $U_{p-1} \wedge Q^2_{i_p p}$ is integral is then a consequence of Lemma EC.5. Now consider $(U_{p-1} \wedge Q^2_{i_p p}) \wedge Q^1_p$. Note that the common variables between $(U_{p-1} \wedge Q^2_{i_p p})$ and Q^1_p are $\{\lambda_\ell^p\}_\ell$. Further note that Q^1_p is integral from Proposition EC.4 and, from above, we have that $(U_{p-1} \wedge Q^2_{i_p p})$ is integral. It is clear that $proj_{\{\lambda_\ell^p\}} Q^1_p = \Delta$. This follows from Lemma EC.4 and Lemma EC.6 (i). Likewise, $proj_{\{\lambda_\ell^p\}} (U_{p-1} \wedge Q^2_{i_p p}) = \Delta$. This follows from Lemma EC.4 once again and Lemma EC.7 (i). The conclusion that $(U_{p-1} \wedge Q^2_{i_p p}) \wedge Q^1_p = U_p$ is integral is then a consequence of Lemma EC.5. □

Appendix D: Proof of Theorem 2

Before proving Theorem 2, we remark that the size of the LP (12) can be slightly reduced.

REMARK EC.1. One can reduce the size of (12) by changing ($12c^z_{j_z}$), which enforces consistency in all bags $z \in T$ containing (SI_{j_z}, S_{j_z}). Alternatively, one can assign a unique z_j to each node j such that $(SI_j, S_j) \subseteq X_{z_j}$ (whose existence is guaranteed by the definition of a tree decomposition), and only impose

$$r^j_{k_j \ell_j} = \sum_{K_{z_j} \setminus k_j} \sum_{L_{z_j} \setminus \ell_j} s^{z_j}_{K_{z_j} L_{z_j}}, \ \forall K_{z_j}, L_{z_j}, \ \forall j \in V.$$

We now move onto the proof of Theorem 2, with a structure following Figure EC.2. While the proofs of Theorems 1 and 2 have a very similar structure, they differ on the use of tree decomposition notation for the latter.

Showing Step 1

PROPOSITION EC.5. *Problems* (12) *and* (2) *are equivalent, provided that the variables appearing in* (12) *are constrained to be integers.*

Proof. Let S_j, SI_j be a solution to (2). We let:

$$\begin{aligned} s^z_{K_z L_z} &= 1_{S_{i_1}=k_{i_1},\ldots,S_{i_{|I_z|}}=k_{i_{|I_z|}},SI_{j_1}=\ell_{j_1},\ldots,SI_{j_{|J_z|}}=\ell_{j_{|J_z|}}} \forall K_z \in \mathcal{D}_{K_z}, L_z \in \mathcal{D}_{L_z}, \forall z \in T \\ r^{j_z}_{k_{j_z}\ell_{j_z}} &= 1_{S_{j_z}=k_{j_z},SI_{j_z}=\ell_{j_z}}, \forall k_{j_z}, \ell_{j_z}, z \in T. \end{aligned} \tag{EC.3}$$

We argue that (EC.3) is feasible for (12). Constraints ($12c^z_{j_z}$) hold by definition, as does ($12e^z$). Furthermore, ($12d^z$) hold, as both the left hand and right hand side of the equalities sum up to the indicator functions of all variables (S_i, SI_j) contained in the overlaps of the bags z and $p(z)$. Constraints ($12a_j$) follow from ($2a_j$) in an identical way to how ($4a_j$) follow from ($2a_j$) in the proof of Proposition EC.1. Likewise, constraints ($12b^z_{i_z j_z}$) hold due to the definition of $\mathcal{S}_{i_z j_z}$. Let $z \in T$ and let $(i_z, j_z) \in (I_z, J_z) \cap E$. As constraint ($2b_{ij}$) enforces that SI_{j_z} cannot be smaller than S_{i_z}, it must be the case that $s^z_{K_z L_z} = 0$ for any values $(K_z, L_z) \in \mathcal{S}_{i_z j_z}$. Finally, note that the objective values of (12) and (2) are equal: this follows from the same argument as the one given in the proof of Proposition EC.1, remarking that for any $j \in V$, the variables SI_j and S_j must appear conjointly in at least one bag z of the tree T by definition.

Conversely, let $\{r^j_{k\ell}\}$ and $\{s^z_{K_z L_z}\}$ be integer solutions to (12). Define (SI_j, S_j) for $j \in V$ as done in (13). We show that this is a feasible solution to (2). As a consequence of ($12e^z$) and ($12c^z_{j_z}$), it follows that $\{r^j_{k\ell}\}_{k,\ell} \in \Delta$. This, together with integrality of the solution, implies that there exists $\ell_j \in \{0, \ldots, M_j\}$ and $k_j \in \{0, \ldots, \Gamma_j\}$, such that $r^j_{k_j \ell_j} = 1$ with $r^j_{k\ell} = 0$ for any $(k, \ell) \neq (k_j, \ell_j)$, which in turn implies that $S_j = k_j$ and $SI_j = \ell_j$ given their definitions. The fact that ($2a_j$) holds and that the objective value of (2) and (12) are equal then follows identically to the proof of Proposition EC.1. To show that ($2b_{ij}$) holds, let $(i, j) \in E$. By definition of $\mathcal{T}$, it must be the case that (SI_j, S_i) appear conjointly in a bag z of the tree T. Let i_z and j_z be their indexes in this bag. One can define:

$$q^{i_z j_z}_{k_{i_z}\ell_{j_z}} = \sum_{K_z \setminus k_{i_z}} \sum_{L_z \setminus \ell_{j_z}} s^z_{K_z L_z}, \forall k_{i_z}, \ell_{j_z}.$$

Constraint ($12b^z_{i_z j_z}$) then implies that $q^{i_z j_z}_{k_{i_z}\ell_{j_z}} = 0, \forall (k_{i_z}, \ell_{i_z}) \in \{0, \ldots, \Gamma_{i_z}\} \times \{0, \ldots, M_{j_z}\}$, such that $k_{i_z} > \ell_{j_z}$. The result follows using the same argument as the one that appears in the proof of Proposition EC.1. □

Showing Step 2a

We now introduce a new linear program with "overlap" variables. This LP can be viewed as the analog of (EC.2): there, $\{\lambda^j_\ell\}_\ell$ and $\{\gamma^j_k\}_k$ play the role of overlap variables. Recall the definition of $\hat{I}_z$ and $\hat{J}_z$ as in (11). We let

$$\hat{K}_z = (k_{i_1}, \ldots, k_{i_{|\hat{I}_z|}}) \text{ for } i_1, \ldots, i_{|\hat{I}_z|} \in \hat{I}_z \text{ and } \hat{L}_z = (\ell_{j_1}, \ldots, \ell_{j_{|\hat{J}_z|}}) \text{ for } j_1, \ldots, j_{|\hat{J}_z|} \in \hat{J}_z, \text{ with}$$
$$\hat{K}_z \in \hat{\mathcal{D}}_{K_z} := \{0, \ldots, \Gamma_{i_1}\} \times \ldots \times \{0, \ldots, \Gamma_{i_{|\hat{I}_z|}}\} \text{ and } \hat{L}_z \in \hat{\mathcal{D}}_{L_z} := \{0, \ldots, M_{j_1}\} \times \ldots \times \{0, \ldots, M_{j_{|\hat{J}_z|}}\}.$$

For each $z \in T$ (excluding the root node), we define additional decision variables corresponding to the overlap variables $\{w^z_{\hat{K}_z \hat{L}_z}\}_{(\hat{K}_z, \hat{L}_z) \in \hat{\mathcal{D}}_{K_z} \times \hat{\mathcal{D}}_{L_z}}$, which track the values $k_i \in \{0, \ldots, \Gamma_i\}$ and $\ell_j \in \{0, \ldots, M_j\}$ taken on by variables S_i (for $i \in \hat{I}_z$) and variables SI_j (for $j \in \hat{J}_z$) present in the overlap $X_z \cap X_{p(z)}$. We can now introduce a new linear program:

$$\min \quad \sum_{j \in V} \sum_{\ell=0}^{M_j} \sum_{k=0}^{\Gamma_j} f_j(\ell - k) r^j_{k\ell} \tag{EC.4}$$

$$\text{s.t.} \quad r^j_{k\ell} \leq \left\lfloor \frac{T_j}{k - \ell} \right\rfloor, \ \forall k = 0, \ldots, \Gamma_j, \ \ell = 0, \ldots, M_j, \ k > \ell, \ j \in V, \tag{EC.4a_j}$$

$$s^z_{K_z L_z} = 0, \ \forall (K_z, L_z) \in \mathcal{S}_{i_z j_z}, \forall (i_z, j_z) \in (I_z \times J_z) \cap E, \ z \in T, \tag{EC.4$b^z_{i_z j_z}$}$$

$$r_{k\ell}^{j_z} = \sum_{K_z \setminus k} \sum_{L_z \setminus \ell} s_{K_z L_z}^{z}, \; \forall (K_z, L_z) \in \mathcal{D}_{K_z} \times \mathcal{D}_{L_z}, \; \forall j_z \in I_z \cap J_z, \; \forall z \in T_1, \qquad (EC.4c_{j_z}^{z})$$

$$w_{\hat{K}_z \hat{L}_z}^{z} = \sum_{K_z \setminus \{k_i\}_{i \in \hat{I}_z}} \sum_{L_z \setminus \{\ell_j\}_{j \in \hat{J}_z}} s_{K_z L_z}^{z} \forall (\hat{K}_z, \hat{L}_z) \in \hat{\mathcal{D}}_{K_z} \times \hat{\mathcal{D}}_{L_z}, \; \forall \; z \in T \setminus \{root\}, \qquad (EC.4d^{z})$$

$$w_{\hat{K}_z \hat{L}_z}^{z} = \sum_{K_{p(z)} \setminus \{k_i\}_{i \in \hat{I}_z}} \sum_{L_{p(z)} \setminus \{\ell_j\}_{j \in \hat{J}_z}} s_{K_{p(z)} L_{p(z)}}^{p(z)} \forall (\hat{K}_z, \hat{L}_z) \in \hat{\mathcal{D}}_{K_z} \times \hat{\mathcal{D}}_{L_z}, \qquad (EC.4e^{z})$$

$$\left\{ s_{K_z L_z}^{z} \right\}_{(K_z, L_z) \in \mathcal{D}_{K_z} \times \mathcal{D}_{L_z}} \in \Delta, \; \forall z \in T, \qquad (EC.4f^{z})$$

where the decision variables are

$$\left\{ r_{k\ell}^{j} \right\}_{k=0,\dots,\Gamma_j, \; \ell=0,\dots,M_j} \text{ for } j \in V, \; \left\{ w_{\hat{K}_z \hat{L}_z}^{z} \right\}_{(\hat{K}_z, \hat{L}_z) \in \hat{\mathcal{D}}_{K_z} \times \hat{\mathcal{D}}_{L_z}} \text{ for } z \in T \setminus \{root\}, \left\{ s_{K_z L_z}^{z} \right\}_{(K_z, L_z) \in \mathcal{D}_{K_z} \times \mathcal{D}_{L_z}} \text{ for } z \in T.$$

We denote by P_{lin}^{ω} its feasible set and show that ($EC.4$) is equivalent to (12).

PROPOSITION EC.6. *The LP ($EC.4$) is equivalent to (12). Furthermore, if the feasible set of ($EC.4$) is integral, then the feasible set of (12) is integral.*

Proof. The proof is identical to that of Proposition EC.2 noting that the new variables $\{w_{\hat{K}_z \hat{L}_z}^{z}\}$ introduced are obtained as marginals of $\{s_{K_z L_z}^{z}\}$.

Showing Step 2b

Recall the definitions of T_1 and T_2 as given in (9). Similarly to Appendix C, we now define Q_z^1 for $z \in T_1$ and Q_z^2 for $z \in T_2$. We use the notation $ch(z)$ to mean the children of node z in T. First, for $z \in T_1$, let

$$Y_z^1 = \left\{ \left\{ r_{k\ell}^{j} \right\}_{k,\ell} \text{ for } j \in I_z \cap J_z, \; \left\{ w_{\hat{K}_v \hat{L}_v}^{v} \right\}_{\hat{K}_v, \hat{L}_v} \text{ for } v \in \{z, ch(z)\}, \; \left\{ s_{K_z L_z}^{z} \right\}_{K_z, L_z} \right\}$$

and define:

$$Q_z^1 = \left\{ Y_z^1 \mid (EC.4a_j)_{j \in I_z \cap J_z}, (EC.4b_{ij}^{z})_{(i,j) \in (I_z, J_z) \cap E}, (EC.4c_{j_z}^{z}), (EC.4d^{v})_{v \in \{z, ch(z)\}}, (EC.4e^{v})_{v \in \{z, ch(z)\}}, (EC.4f^{z}) \right\}.$$

Second, for $z \in T_2$, let

$$Y_z^2 = \left\{ \left\{ w_{\hat{K}_v \hat{L}_v}^{v} \right\}_{\hat{K}_v, \hat{L}_v} \text{ for } v \in \{z, ch(z)\}, \; \left\{ s_{K_z L_z}^{z} \right\}_{K_z, L_z} \right\}$$

and define

$$Q_z^2 = \left\{ Y_z^2 \mid (EC.4b_{ij}^{z})_{(i,j) \in (I_z, J_z) \cap E}, (EC.4d^{v})_{v \in \{z, ch(z)\}}, (EC.4e^{v})_{v \in \{z, ch(z)\}}, (EC.4f^{z}) \right\}.$$

Recalling that P_{lin}^{ω} is the feasible set of ($EC.4$), it is immediate to establish, using similar arguments to Proposition EC.3, that:

PROPOSITION EC.7. *We have* $P_{lin}^{\omega} = \left(\wedge_{z \in T_1} Q_z^1 \right) \wedge \left(\wedge_{z \in T_2} Q_z^2 \right)$.

Showing Step 3

The proof of integrality of P_{lin}^{ω} requires a few intermediary propositions (see Figure EC.2). Our first proposition is useful for both the base and induction steps.

PROPOSITION EC.8. *The polytope Q_z^1 is integral for any $z \in T_1$ and the polytope Q_z^2 is integral for any $z \in T_2$.*

Proof. Let $z \in T_2$. If we drop the constraints ($EC.4b^z_{ij}$) for $(i,j) \in (I_z, J_z) \cap E$ in Q^2_z, then the resulting polytope is a simplex with binary vertices, following Lemma EC.1. Adding back the constraints, we obtain a simplex with binary vertices once again from Lemma EC.2. Thus, Q^2_z is integral.

Suppose now that $z \in T_1$. Similarly, if we drop the constraints ($EC.4a_j$) in Q^1_z for $j \in I_z \cap J_z$ and ($EC.4b^z_{i_z j_z}$) for $(i_z, j_z) \in (I_z, J_z) \cap E$, we obtain a simplex with binary vertices following Lemma EC.1. Adding back the constraints ($EC.4a_j$) for $j \in I_z \cap J_z$, we obtain once again a simplex with binary vertices, following Lemma EC.3, and this happens once again when we add back the constraints ($EC.4b^z_{i_z j_z}$) for $(i_z, j_z) \in (I_z, J_z) \cap E$, following Lemma EC.2. Thus, Q^1_z is integral. □

We now show Lemma EC.9, which is an analog of Lemmas EC.6 and EC.7. Unlike the tree case, where things were quite simple, showing Lemma EC.9 "from scratch" is a lot more involved due to the notation. Thus, instead of dealing with the binary variables r, w, s, which are quite cumbersome, we show a version of Lemma EC.9 using the integer variables SI, S. This is more straightforward, but requires the introduction of new notation and a proof that operating in the "binary space" is equivalent to operating in the "integer space". Before showing this result and Lemma EC.9, we first introduce an ordering of the leaves of the tree T involved in the tree decomposition of G' and some notation. Indeed, the proof of Lemma EC.9 requires an ordering of the leaves of T with a specific property—this is similar to the tree case, except that we were directly labeling G there, as opposed to T. As before, we use the notation $p(z)$ to denote the unique parent of a node z in T.

LEMMA EC.8. *Let G' be an intersection graph as defined in Definition 2, and let $\mathcal{T}' = (T, \{X_z\}_{z \in T})$ be a tree decomposition of G' of width ω. There exists a labeling $(1, \ldots, |T|)$ of the nodes of T such that $(X_1 \cup \ldots \cup X_{i-1}) \cap X_i = X_{p(i)} \cap X_i$ for all $i = 2, \ldots, |T| - 1$. In particular, the overlap is of size at most ω.*

Proof. Pick a node of T to be the root and then label the nodes using, e.g., the level-order traversal. This ensures that the labels of the parents are before those of the children. As a child in a tree has a unique parent, node i is connected via exactly one edge to the subtree generated by $(1, \ldots, i)$, through its parent. The properties of a tree decomposition give us that $(X_1 \cup \ldots \cup X_{i-1}) \cap X_i = X_{p(i)} \cap X_i$. As one can always find a tree decomposition with distinct bags (otherwise, they can be merged), the overlap is of size at most ω, as each bag contains $\omega + 1$ variables. □

Let $(1, \ldots, |T|)$ be an ordering of the nodes of T as in Lemma EC.8. We are now ready to introduce the additional notation required to operate in the "integer space" (see Table EC.2). Recall the definition of $\hat{I}_z$ and $\hat{J}_z$ as given in (11).

Note that in Table EC.2, we use the notation $\{X \mid \text{constraints}(X, Y)\}$, where X is a set of variables and constraints(X, Y) are constraints involving X and another set of variables Y, to mean the set $\{X \mid \exists Y \text{ constraints}(X, Y)\}$. The first table describes polytopes in the integer space which, when joined together, form P^{ω}_{lin}. The second table does the same, but in the binary space. The following proposition shows the equivalence between operating in the two spaces:

PROPOSITION EC.9. *We have:*

(i) For any $i \in \{1, 2\}$ and any $z \in T$, there exists a point in $\tilde{R}^i_z$ if and only if there exists an integral point in $\tilde{Q}^i_z$.

(ii) For any $z \in T$, there exists a point in $\left(\wedge_{z' \in \{1,\ldots,z\} \cap T_1} R^1_z\right) \wedge \left(\wedge_{z' \in \{1,\ldots,z\} \cap T_2} R^2_z\right)$ if and only if there exists an integral point in $\left(\wedge_{z' \in \{1,\ldots,z\} \cap T_1} Q^1_z\right) \wedge \left(\wedge_{z' \in \{1,\ldots,z\} \cap T_2} Q^2_z\right)$.

Table EC.2 Sets of variables and polytopes of interest in the proof of Step 2c of Theorem 2.

	Integer Space (SI, S)
Bags in T_1	$X_z^1 = \{\{S_i\}_{i\in I_z}, \{SI_j\}_{j\in J_z}\}, \quad I_z \cap J_z \neq \emptyset$ $R_z^1 = \{X_z^1 \mid (2a_j)_{j\in I_z\cap J_z}, (2b_{ij})_{(i,j)\in I_z\times J_z\cap E}, (2c_i)_{i\in I_z}, (2d_j)_{j\in J_z}\}$
Bags in T_2	$X_z^2 = \{\{S_i\}_{i\in I_z}, \{SI_j\}_{j\in J_z}\}, \quad I_z \cap J_z = \emptyset$ $R_z^2 = \{X_z^2 \mid (2b_{ij})_{(i,j)\in I_z\times J_z\cap E}, (2c_i)_{i\in I_z}, (2d_j)_{j\in J_z}\}$
Overlaps - Case 1 $\hat{I}_z \cap \hat{J}_z = \emptyset$	$\tilde{X}_z^1 = \{\{S_i\}_{i\in \hat{I}_z}, \{SI_j\}_{j\in \hat{J}_z}\}$ $\tilde{R}_z^1 = \{\tilde{X}_z^1 \mid (2b_{ij})_{(i,j)\in(\hat{I}_z\times\hat{J}_z)\cap E}, (2c_i)_{i\in\hat{I}_z}, (2d_j)_{j\in\hat{J}_z}\}$
Overlaps - Case 2 $\hat{I}_z \cap \hat{J}_z \neq \emptyset$	$\tilde{X}_z^2 = \{\{S_i\}_{i\in \hat{I}_z}, \{SI_j\}_{j\in \hat{J}_z}\}$ $\tilde{R}_z^2 = \{\tilde{X}_z^2 \mid (2a_j)_{j\in\hat{I}_z\cap\hat{J}_z}, (2b_{ij})_{(i,j)\in(\hat{I}_z\times J_z)\cap E}, (2c_i)_{i\in\hat{I}_z}, (2d_j)_{j\in\hat{J}_z}\}$

	Binary Space (r, w, s)
Bags in T_1	$Y_z^1 = \left\{\left\{r^j_{k\ell}\right\}_{k,l} \text{ for } j \in I_z \cap J_z,\ \left\{w^v_{\hat{K}_v\hat{L}_v}\right\}_{\hat{K}_v,\hat{L}_v} \text{ for } v \in \{z, ch(z)\},\ \left\{s^z_{K_zL_z}\right\}_{K_z,L_z}\right\}$ $Q_z^1 = \Big\{Y_z^1 \mid (EC.4a_j)_{j\in I_z\cap J_z}, (EC.4b^z_{ij})_{(i,j)\in(I_z,J_z)\cap E}, (EC.4c^z_{j_z}), (EC.4d^v)_{v\in\{z,ch(z)\}}, (EC.4e^v)_{v\in\{z,ch(z)\}}, (EC.4f^z),\Big\}$
Bags in T_2	$Y_z^2 = \left\{\left\{w^v_{\hat{K}_v\hat{L}_v}\right\}_{\hat{K}_v,\hat{L}_v} \text{ for } v \in \{z, ch(z)\},\ \left\{s^z_{K_zL_z}\right\}_{K_z,L_z}\right\}$ $Q_z^2 = \Big\{Y_z^2 \mid (EC.4b^z_{ij})_{(i,j)\in(I_z,J_z)\cap E}, (EC.4d^v)_{v\in\{z,ch(z)\}}, (EC.4e^v)_{v\in\{z,ch(z)\}}, (EC.4f^z)\Big\}$
Overlaps - Case 1 $\hat{I}_z \cap \hat{J}_z = \emptyset$	$\tilde{Y}_z^1 = \left\{\left\{w^z_{\hat{K}_z\hat{L}_z}\right\}_{\hat{K}_z,\hat{L}_z}\right\}$ $\tilde{Q}_z^1 = \Big\{\tilde{Y}_z^1 \mid (EC.4b^z_{ij})_{(i,j)\in\hat{I}_z\times\hat{J}_z\cap E}, (EC.4d^z), (EC.4e^z), (EC.4f^v)_{v\in\{z,p(z)\}}\Big\}$
Overlaps - Case 2 $\hat{I}_z \cap \hat{J}_z \neq \emptyset$	$\tilde{Y}_z^2 = \left\{\left\{r^j_{k\ell}\right\}_{k,\ell} \text{ for } j \in \hat{I}_z \cap \hat{J}_z, \left\{w^z_{\hat{K}_z\hat{L}_z}\right\}_{\hat{K}_z,\hat{L}_z}\right\}$ $\tilde{Q}_z^2 = \Big\{\tilde{Y}_z^2 \mid (EC.4a_j)_{j\in\hat{I}_z\cap\hat{J}_z}, (EC.4b^z_{ij})_{(i,j)\in\hat{I}_z\times\hat{J}_z\cap E}, (EC.4c^z_{j_z})_{j_z\in\hat{I}_z\cap\hat{J}_z}, (EC.4d^z), (EC.4e^z), (EC.4f^v)_{v\in\{z,p(z)\}}\Big\}$

Proof. We show (i) first. For $i = 1$, if there exists a point in $\tilde{R}_z^1$, simply set

$$w^z_{\hat{K}_z\hat{L}_z} = \mathbf{1}_{S_i=k_i, i\in\hat{I}_z, SI_j=\ell, j\in\hat{J}_z} \text{ and } s^v_{K_vL_v} = \mathbf{1}_{S_i=k_i, i\in I_v, SI_j=\ell, j\in J_v} \text{ for } v \in \{z, p(z)\}.$$

These solutions are integral and it is easy to check that they belong to $\tilde{Q}_z^1$. Conversely, if $\tilde{Q}_z^1$ is nonempty and integral, then constraint $(EC.4f^z)$ for $v \in \{z, p(z)\}$ combined to $(EC.4d^z)$ and $(EC.4e^z)$ implies that $\left\{w^z_{\hat{K}_z\hat{L}_z}\right\}_{\hat{K}_z,\hat{L}_z} \in \Delta$. Using the integrality assumption, this means that there exist $\hat{K}_z'$ and $\hat{L}_z'$ such that $w^z_{\hat{K}_z'\hat{L}_z'} = 1$ with $w^z_{\hat{K}_z\hat{L}_z} = 0$ for all $(\hat{K}_z, \hat{L}_z) \neq (\hat{K}_z', \hat{L}_z')$. Setting $S_i = k_i'$ for $i \in \hat{I}_z$ and $SI_j = \ell'$ for $j \in \hat{J}_z$, we have that $(EC.4b^z_{ij})$ implies $(2b_{ij})$ for $(i, j) \in \hat{I}_z \times \hat{J}_z \cap E$. The constraints $(2c_j)$ for $i \in \hat{I}_z$ and $(2d_j)$ for $j \in \hat{J}_z$ follow by definition. When $i = 2$, if $\tilde{R}_z^2$ is nonempty and integral, we simply define additional integral variables $r^j_{k\ell} = \mathbf{1}_{S_j=k, SI_j=\ell}$ for all $j \in \hat{I}_z \cap \hat{J}_z$ and $\tilde{Q}_z^2$ is nonempty. The converse holds immediately by noting that for $j_z \in \hat{I}_z \cap \hat{J}_z$, constraint $(EC.4c^z_{j_z})$ together with integrality and the previous argument implies that $r^{j_z}_{k'\ell'}$ must be equal to $\mathbf{1}_{S_{j_z}=k', SI_{j_z}=\ell'}$ and zero elsewhere. Constraint $(2a_j)$ then follows from $(EC.4a_j)$ for any $j \in \hat{I}_z \cap \hat{J}_z$.

We now show (ii). If $\left(\wedge_{z'\in\{1,\dots,z\}\cap T_1} R^1_z\right) \wedge \left(\wedge_{z'\in\{1,\dots,z\}\cap T_2} R^2_z\right)$ is non-empty, then take

$$r^{j_{z'}}_{k\ell} = \mathbf{1}_{S_{j_{z'}}=k, SI_{j_{z'}}=\ell},\ \forall k,\ell,\ \forall j_{z'} \in I_{z'} \cap J_{z'},\ \forall z' \in \{1,\dots,z\} \cap T_1,$$
$$w^{z'}_{\hat{K}_{z'}\hat{L}_{z'}} = \mathbf{1}_{S_i=k_i, i\in \hat{I}_{z'}, SI_j=\ell, j\in \hat{J}_{z'}},\ \forall \hat{K}_{z'}\hat{L}_{z'},\ \forall z' \in \{2,\dots,z\}$$
$$s^{z'}_{K_{z'}L_{z'}} = \mathbf{1}_{S_i=k_i, i\in I_{z'}, SI_j=\ell, j\in J_{z'}},\ \forall K_{z'}, L_{z'},\ \forall z' = 1,\dots,z.$$

These are binary variables and, thus, integers. By construction, it is easy to see that $(EC.4c^{z'}_{j_{z'}})$, $(EC.4d^{z'})$, $(EC.4e^{z'})$, and $(EC.4f^{z'})$ hold for any $z' \in \{1,\dots,z\}$. For constraint $(EC.4a_{j_{z'}})$, the argument is identical to the proof of Proposition EC.1 for the tree case and holds for any $z' \in \{1,\dots,z\}$. Let $z' \in \{1,\dots,z\}$. For $(EC.4b^{z'}_{i_{z'}j_{z'}})$, suppose that $(2b_{i_{z'}j_{z'}})$ holds for $(S_{i_{z'}}, SI_{j_{z'}})$. Moreover, suppose that $SI_{j_{z'}} = \bar{\ell}$ and $S_{i_{z'}} = \bar{k}$ for some $\bar{k}, \bar{\ell}$. As $SI_{j_{z'}} - S_{i_{z'}} \geq 0$, we must have that $\bar{\ell} \geq \bar{k}$. Thus, it cannot be that $\bar{k} > \bar{\ell}$ and so $s^{z'}_{K_{z'}L_{z'}} = 0$ when $\bar{k} > \bar{\ell}$. Thus, $(EC.4b^{z'}_{i_{z'}j_{z'}})$ holds given the definition of $\mathcal{S}_{ij}$ in (10). This implies that there exists an integral point in $\left(\wedge_{z'\in\{1,\dots,z\}\cap T_1} Q^1_z\right) \wedge \left(\wedge_{z'\in\{1,\dots,z\}\cap T_2} Q^2_z\right)$.

Now suppose that $\left(\wedge_{z'\in\{1,\dots,z\}\cap T_1} Q^1_z\right) \wedge \left(\wedge_{z'\in\{1,\dots,z\}\cap T_2} Q^2_z\right)$ contains an integral point. As $Y^1_{z'}$ for $z' \in T_1$ and $Y^2_{z'}$ for $z' \in T_2$ are integral, and given the constraints of $Q^1_{z'}$ and $Q^2_{z'}$, it follows that all variables are binary. Thus, for any $z' \in \{1,\dots,z\}$, we can define:

$$S_{i_{z'}} = \sum_{k=0}^{\Gamma_{i_{z'}}} k \cdot \gamma^{i_{z'}}_k \quad \text{and } SI_{j_{z'}} = \sum_{\ell=0}^{M_{j_{z'}}} \ell \cdot \lambda^{j_{z'}}_\ell,$$

where

$$\gamma^{i_{z'}}_k = \sum_{K_{z'}\setminus k_{i_{z'}}} \sum_{L_{z'}} s^{z'}_{K_{z'}L_{z'}},\ \forall k = 0,\dots,\Gamma_{i_{z'}},\ \text{and } \lambda^{j_{z'}}_\ell = \sum_{K_{z'}} \sum_{L_{z'}\setminus \ell_{j_{z'}}} s^{z'}_{K_{z'}L_{z'}},\ \forall \ell = 0,\dots,M_{j_{z'}}. \tag{EC.5}$$

Note that constraints $(EC.4d^{z'})$ and $(EC.4e^{z'})$ enforce consistency of these variables across bags $z' \in \{1,\dots,z\}$: that is, as z' changes, $S_{i_{z'}}$ and $SI_{j_{z'}}$ do not. The fact that $(2c_{j_{z'}})$ and $(2d_{j_{z'}})$ hold is immediate for any $z' \in \{1,\dots,z\}$. Now, let $z' \in \{1,\dots,z\} \cap T_1$. For constraint $(2a_{j_{z'}})$, constraints $(EC.4f^{z'})$ and $(EC.4c^{z'}_{j_{z'}})$ imply that $\{r^{j_{z'}}_{k\ell}\}_{k,\ell} \in \Delta$ for any $j_{z'} \in I_{z'} \cap J_{z'}$ and $z' \in \{1,\dots,z\}$. As all variables are integer, it follows that $\{r^{j_{z'}}_{k\ell}\}_{k,\ell}$ is binary. There exist $\hat{\ell}_{j_{z'}} \in \{0,\dots,M\}$ and $\hat{k}_{j_{z'}} \in \{0,\dots,\Gamma_{j_{z'}}\}$, such that $r^{j_{z'}}_{\hat{k}_{j_{z'}}\hat{\ell}_{j_{z'}}} = 1$ with $r^j_{k\ell} = 0$ for any $(k,\ell) \neq (\hat{k}_{j_{z'}}, \hat{\ell}_{j_{z'}})$. Using $(EC.4c^{z'}_{j_{z'}})$ in the definition (EC.5), we get

$$\gamma^{j_{z'}}_k = \sum_\ell r^{j_{z'}}_{k\ell},\ \lambda^{j_{z'}}_\ell = \sum_k r^{j_{z'}}_{k\ell},$$

which implies that $S_{j_{z'}} = \hat{k}_{j_{z'}}$ and $SI_{j_{z'}} = \hat{\ell}_{j_{z'}}$. If $\hat{k}_{j_{z'}} \leq \hat{\ell}_{j_{z'}}$, then $S_{j_{z'}} - SI_{j_{z'}} = \hat{k}_{j_{z'}} - \hat{\ell}_{j_{z'}} \leq T_{j_{z'}}$ trivially holds, as $T_{j_{z'}} \geq 0$. Now suppose that $\hat{k}_{j_{z'}} > \hat{\ell}_{j_{z'}}$. Constraint $(EC.4a_{j_{z'}})$ implies that $r_{\hat{k}_{j_{z'}}\hat{\ell}_{j_{z'}}} \leq T_{j_{z'}}/(\hat{k}_{j_{z'}} - \hat{\ell}_{j_{z'}})$, i.e., $(\hat{k}_{j_{z'}} - \hat{\ell}_{j_{z'}}) = S_{j_{z'}} - SI_{j_{z'}} \leq T_{j_{z'}}$, which is $(2a_{j_{z'}})$. Now, let $z' \in \{1,\dots,z\}$. From (EC.5), we have that:

$$SI_{j_{z'}} - S_{i_{z'}} = \sum_{\ell=0}^{M_{j_{z'}}} \ell \cdot \lambda^{j_{z'}}_\ell - \sum_{k=0}^{\Gamma_{i_{z'}}} k \cdot \gamma^{i_{z'}}_k = \sum_{K_{z'}\setminus k_{i_{z'}}} \sum_{L_{z'}\setminus \ell_{j_{z'}}} (\ell - k) s^{z'}_{K_{z'}L_{z'}}.$$

From the definition of $\mathcal{S}_{i_{z'}j_{z'}}$, constraint $(EC.4b^{z'}_{i_{z'}j_{z'}})$ implies that if $k > \ell$, $s^{z'}_{K_{z'}L_{z'}} = 0$. Thus, $SI_{j_{z'}} - S_{i_{z'}} \geq 0$, which is $(2b_{i_{z'}j_{z'}})$. $\square$

LEMMA EC.9. *Let $\tilde{Q}^i_z$ for $i \in \{1,2\}$ and Q^i_z for $i \in \{1,2\}$ be the sets defined in Table EC.2. We have:*

(i) For $z \in T_1$, if $\tilde{Q}^1_z$ or $\tilde{Q}^2_z$ contains an integral point, then Q^1_z contains an integral point.

(ii) For $z \in T_2$, if $\tilde{Q}_z^1$ contains an integral point, then Q_z^2 contains an integral point.

(iii) For $z \in T$, if either $\tilde{Q}_z^1$ or $\tilde{Q}_z^2$ contains an integral point, then $\left(\wedge_{z' \in \{1,\dots,p(z)\} \cap T_1} Q_{z'}^1\right) \wedge \left(\wedge_{z' \in \{1,\dots,p(z)\} \cap T_2} Q_{z'}^2\right)$ contains an integral point.

Proof. We show instead the following statements:

(i) For $z \in T_1$, if $\tilde{R}_z^1$ or $\tilde{R}_z^2$ is nonempty, then R_z^1 is nonempty.

(ii) For $z \in T_2$, if $\tilde{R}_z^1$ is nonempty, then R_z^2 is nonempty.

(iii) For $z \in T$, if $\tilde{R}_z^1$ or $\tilde{R}_z^2$ is nonempty, then $\left(\wedge_{z' \in \{1,\dots,z\} \cap T_1} R_{z'}^1\right) \wedge \left(\wedge_{z' \in \{1,\dots,z\} \cap T_2} R_{z'}^2\right)$ is nonempty.

The proposition follows immediately from Proposition EC.9. Note that the proof of (iii) is exactly the same as the one in the proof of Proposition EC.7 so we only show (i) and (ii).

To show (i), let $z \in T_1$ and assume that $\tilde{R}_z^1$ is nonempty first. Thus, the values of $\{S_i\}_{i \in \hat{I}_z}$ and $\{SI_j\}_{j \in \hat{J}_z}$ are fixed. Take $S_i = 0$ for all $i \in I_z, i \notin \hat{I}_z$ and let $SI_j = \max\{\max_{i \in \hat{I}_z, i \neq j}\{S_i\}, S_j - T_j\}$ for $j \in J_z, j \notin \hat{J}_z$. By construction, such a point is in R_z^1. The same construction and argument holds when $\tilde{R}_z^2$ is nonempty. For (ii), let $z \in T_2$ and assume that $\tilde{R}_z^1$ is nonempty. Then, as before, the values of $\{S_i\}_{i \in \hat{I}_z}$ and $\{SI_j\}_{j \in \hat{J}_z}$ are fixed. Take $S_i = 0$ for all $i \in I_z, i \notin \hat{I}_z$ and let $SI_j = \max_{i \in \hat{I}_z, i \neq j}\{S_i\}$ for $j \in J_z, j \notin \hat{J}_z$. Such a point is in R_z^2. □

We now have all the elements to show Theorem 2:

Proof of Theorem 2. It is enough to show that the feasible set P of $(EC.4)$ is integral. Indeed, Proposition EC.6 then implies that the feasible set of (12) is integral. From Proposition EC.5, it follows that solving $(EC.4)$ is equivalent to showing (2), which gives us Theorem 2. To show that P is integral, let $(1, \dots, |T|)$ be an ordering of the nodes of T as in Lemma EC.8. Define for $z \in (1, \dots, |T|)$ the set

$$U_z^\omega := \left(\wedge_{z' \in \{1,\dots,z\} \cap T_1} Q_z^1\right) \wedge \left(\wedge_{z' \in \{1,\dots,z\} \cap T_2} Q_z^2\right).$$

From Proposition EC.7, it is clear that $U_{|T|}^\omega = P$. We now show by induction on $z \in (1, \dots, |T|)$ that U_z^ω is integral.

Base Case: Suppose that $z = 1$. If $1 \in T_1$, then Proposition EC.8 shows that Q_1^1 is integral; if $1 \in T_2$ then Proposition EC.8 shows that Q_1^2 is integral. Thus, U_1 is integral.

Induction Step: Let $z \in (1, \dots, |T|)$. Suppose that $U_{p(z)}$ is integral and consider U_z^ω. We have $U_z^\omega = U_{p(z)}^\omega \wedge Q_z^1$ if $z \in T_1$ and $U_z^\omega = U_{p(z)}^\omega \wedge Q_z^2$ if $z \in T_2$. We use Lemma EC.5 to show that U_z^ω is integral. The induction hypothesis gives us that $U_{p(z)}$ is integral and Q_z^i is integral from Proposition EC.8 for any $i \in \{1, 2\}$. Thus, it remains to show that the projection of $U_{p(z)}$ and Q_z^i for $i = \{1, 2\}$ onto their common variables is a common simplex. This is what we show now distinguishing following the cases given in Table EC.2. The variables appearing in Q_z^i are Y_z^i for $i \in \{1, 2\}$ and the variables appearing in $U_{p(z)}$ are the variables $\left(\cup_{z' \in \{1,\dots,p(z)\} \cap T_1} Y_{z'}^1\right) \cup \left(\cup_{z' \in \{1,\dots,p(z)\} \cap T_2} Y_{z'}^2\right)$. From Lemma EC.8, the common variables between the two sets are then $Y_{p(z)}^i \cap Y_z^j$ for $i, j \in \{1, 2\}$. A simple check reveals that this intersection can only be $\tilde{Y}_z^1$ (if $\hat{I}_z \cap \hat{J}_z = \emptyset$) or $\tilde{Y}_z^2$ (if $\hat{I}_z \cap \hat{J}_z \neq \emptyset$).

Case 1. Suppose that $\hat{I}_z \cap \hat{J}_z = \emptyset$. We have that

$$proj_{\tilde{Y}_z^1} Q_z^1 = proj_{\tilde{Y}_z^1} U_{p(z)} = D_z \text{ for } z \in T_1 \text{ and } proj_{\tilde{Y}_z^2} Q_z^2 = proj_{\tilde{Y}_z^2} U_{p(z)} = D_z \text{ for } z \in T_2, \tag{EC.6}$$

where

$$D_z = \left\{ \tilde{Y}_z^1 \,\middle|\, w_{\hat{K}_z \hat{L}_z}^z = 0, \forall k_{i_z} > \ell_{j_z} \text{ s.t. } (i_z, j_z) \in E,\ \{w_{\hat{K}_z \hat{L}_z}^z\}_{\hat{K}_z, \hat{L}_z} \in \Delta \right\}.$$

To see why (EC.6) holds, note that any integral point in D_z can be extended to an integral point in $\tilde{Q}_z^1$, which can be extended to an integral point in Q_z^1 or Q_z^2 from Lemma EC.9. Furthermore, any integral point in $\tilde{Q}_z^1$ can be extended to an integral point to $U_{p(z)}$ also from Lemma EC.9. Conversely, if $Y_z^1 \in Q_z^1$ is integral, then it is easy to see that $\tilde{Y}_z^1 \in D_z$ (which will also be integral); likewise, if $Y_z^2 \in Q_z^2$ and is integral, then $\tilde{Y}_z^1 \in D_z$ and is integral. Finally, if $(\cup_{z' \in \{1,\dots,p(z)\} \cap T_1} Y_{z'}^1) \cup (\cup_{z' \in \{1,\dots,p(z)\} \cap T_2} Y_{z'}^2) \in U_{p(z)}$ and is integral, then $\tilde{Y}_z^1 \in D$ and is integral. Thus, (EC.6) follows from Lemma EC.4.

Case 2. Suppose now that $\hat{I}_z \cap \hat{J}_z \neq \emptyset$. We have that

$$proj_{\tilde{Y}_z^2} Q_z^1 = proj_{\tilde{Y}_z^2} U_{p(z)} = D_z, \tag{EC.7}$$

where

$$D_z = \Big\{ \tilde{Y}_z^2 \,\Big|\, r_{k\ell}^{j_z} = \sum_{\hat{K}_z \setminus k_{j_z}} \sum_{\hat{L}_z \setminus \ell_{j_z}} w_{\hat{K}_z \hat{L}_z}^z,\ r_{k\ell}^{j_z} \le \left\lfloor \frac{T_{j_z}}{k-\ell} \right\rfloor,\ \forall k > \ell,\ \forall j_z \in \hat{I}_z \cap \hat{J}_z, w_{\hat{K}_z \hat{L}_z}^z = 0, \forall k_{i_z} > \ell_{j_z} \text{ s.t. } (i_z, j_z) \in E,$$
$$\{w_{\hat{K}_z \hat{L}_z}^z\}_{\hat{K}_z, \hat{L}_z} \in \Delta \Big\}.$$

The set D is a simplex by virtue of Lemmas EC.1, EC.3 and EC.2. One can show once again that (EC.7) holds via Lemma EC.9 and Lemma EC.4. $\square$

Appendix E: Additional proofs and results relating to Sections 4 and 5

E.1. Weakness of the LP for the counterexample in Section 4.1

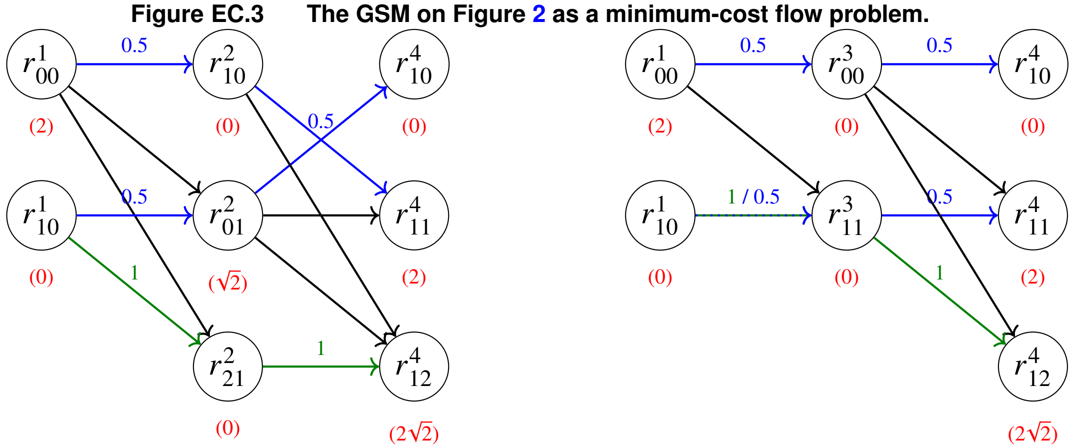
When G is a path, GSM (2), or equivalently LP (4), has a reformulation as a minimum-cost flow problem. This interpretation is particularly useful in clarifying why LP (4) fails to solve the GSM on the diamond network (Figure 2).

This network consists of two overlapping paths: $1 \to 2 \to 4$ and $1 \to 3 \to 4$. Solving the GSM on $1 \to 2 \to 4$ (resp. $1 \to 3 \to 4$) is equivalent to solving a minimum-cost flow problem on the graph in Figure EC.3a (resp. EC.3b), where we send unit flow from source nodes $\{r_{k\ell}^1\}_{k,\ell}$ to sink nodes $\{r_{k\ell}^4\}_{k,\ell}$. Only nodes that are non-zero per constraints ($4a_j$) and edges that are not zeroed out due to constraints ($4b_{ij}$) appear in Figure EC.3. That is, an edge connects $r_{k\ell}^i$ to $r_{k'\ell'}^j$ if there is an arc (i,j) on paths $1 \to 2 \to 4$ or $1 \to 3 \to 4$, and the implied service times $k = S_i \le SI_j = \ell'$. The remainder of the constraints ($4c_j$)-($4f_{ij}$) describe a flow and the cost of passing through each node $r_{k\ell}^j$, indicated in red under the nodes, is given by $f_j(\ell - k)$, the summands of (4)'s objective.

The optimal solution to the GSM (2) on the diamond network is depicted as the green flow in Figure EC.3: it begins at r_{10}^1 and ends at r_{12}^4, following paths $r_{10}^1 \to r_{21}^2 \to r_{12}^4$ and $r_{10}^1 \to r_{11}^3 \to r_{12}^4$, respectively. This flow results in a total cost of $2\sqrt{2}$. Solving the GSM for the diamond network thus amounts to solving a minimum-cost flow problem on the graphs corresponding to the paths $1 \to 2 \to 4$ and $1 \to 3 \to 4$, while ensuring that the source node and destination node of the flow are the same across the graphs, to translate the fact that the paths overlap at 1 and 4 in the diamond. Such a flow is said to be *consistent* and any feasible solution to (2) for the diamond corresponds to a consistent flow.

Now, consider the optimal solution to LP (4) on the diamond network: it is fractional and sends half a unit along each path: $r_{00}^1 \to r_{10}^2 \to r_{11}^4$, $r_{10}^1 \to r_{01}^2 \to r_{10}^4$, $r_{00}^1 \to r_{00}^3 \to r_{10}^4$, and $r_{10}^1 \to r_{11}^3 \to r_{11}^4$ (in blue in Figure EC.3), resulting in a total cost of $2 + \frac{\sqrt{2}}{2}$, strictly less than the optimal GSM cost of $2\sqrt{2}$. While the blue flow does start from the same origin *set* (that is, $\{r_{00}^1, r_{10}^1\}$, with a half quantity for each) and finish in the same destination *set* (that is, $\{r_{10}^4, r_{11}^4\}$, with a half quantity each), it does not have consistent (origin, destination) pairs. For example, it delivers a half unit of flow

Figure EC.3 The GSM on Figure 2 as a minimum-cost flow problem.



(a) The GSM on path $1 \to 2 \to 4$ as a minimum-cost flow problem.

(b) The GSM on path $1 \to 3 \to 4$ as a minimum-cost flow problem.

Note. Values in red beneath each node indicate the cost of flows through that node. Green (resp. blue) arrows indicate the flows corresponding to the optimal solution of (2) (resp. (4)) on the diamond network.

to r^4_{11} starting from point r^1_{00} in Figure EC.3a but from r^1_{10} in Figure EC.3b. Simply put, the blue flow, i.e., the optimal solution to (4), is obtained by interpolating two inconsistent flows, that is, infeasible solutions to (2). This implies that LP (4) admits solutions that do not belong to the convex hull of feasible solution to (2).

This fact can also be seen through a purely algebraic prism. Consider the following inequality:

$$(1 - r^1_{00}) + r^2_{10} + r^3_{00} + r^3_{11} + r^4_{10} \leq 2. \tag{EC.8}$$

We show that this inequality is valid for (2), but invalid for (4). Invalidity for (4) is straightforward, as the optimal solution to (4) has 2.5 on the left-hand side. We use purely algebraic arguments to show that (EC.8) holds for any solution to (2), although it is possible to use Figure EC.3 instead to make the same points through the presence/absence of certain nodes and edges. We start with the case where $r^1_{00} = 0$: we must have $r^1_{10} = 1$, as that is the only remaining option for the values of S_1 and SI_1. It follows that both $r^2_{10} = 0$ and $r^3_{00} = 0$, as constraints $SI_2 - S_1 \geq 0$ and $SI_3 - S_1 \geq 0$ would otherwise be violated. Furthermore, either r^3_{11} or r^4_{10} can be equal to one, but not both, otherwise the constraint $SI_4 - S_3 \geq 0$ would be violated. Thus, (EC.8) holds when $r^1_{10} = 1$. In the case where $r^1_{00} = 1$, at most one of r^2_{10} or r^4_{10} can be equal to one (if both are, $SI_4 - S_2 \geq 0$ is violated). Furthermore, exactly one of r^3_{00} or r^3_{11} is equal to one (if both are zero, then there are no other candidate values for SI_3 and S_3). Thus, (EC.8) holds when $r^1_{00} = 1$.

E.2. Additional illustration of the theory developed in Section 4

We consider here an additional example, the 6-cycle (see Figure EC.4). The intersection graph of the GSM defined over the 6-cycle is provided in Figure EC.5, and its tree decomposition in Figure EC.6. Note that the width of this decomposition (and the treewidth of the intersection graph) is two.

We make explicit the LP from Theorem 2 for this example. For this, we need to introduce one set of variables per bag: for node 2, this is $s^2_{k_1\ell_2\ell_4}$; for node 3, this is $s^3_{\ell_2\ell_4k_4}$; for node 4, this is $s^4_{\ell_2k_2k_4}$; for node 5, this is $s^5_{k_2k_4\ell_5}$; for node 6, this is $s^6_{k_2\ell_3\ell_5}$; for node 7, this is $s^7_{\ell_3\ell_5k_5}$; for node 8, this is $s^8_{\ell_3k_3k_5}$; for node 9, this is $s^9_{\ell_6k_3k_5}$. For nodes 1 and 10, which

Figure EC.4 The 6-cycle considered in Appendix E.2.

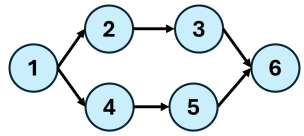


Figure EC.5 The intersection graph for the 6-cycle considered in Appendix E.2.

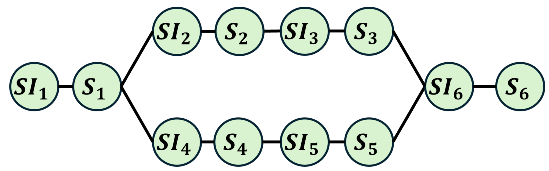
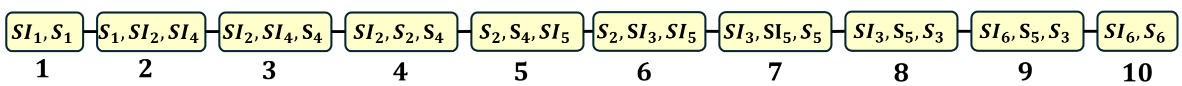


Figure EC.6 The tree decomposition for the intersection graph of the 6-cycle considered in Appendix E.2.



contain same-index pairs (SI_i, S_i), these variables are redundant with $r^1_{k_1\ell_1}$ and $r^6_{k_6\ell_6}$, so they are not needed. We can then write the LP necessary to solve the GSM over the 6-cycle:

$$
\begin{aligned}
\min \quad & \sum_{j=1}^{6}\sum_{\ell=0}^{M_j}\sum_{k=0}^{\Gamma_j} f_j(\ell-k) r^j_{k\ell} \\
\text{s.t.} \quad & r^j_{k\ell} \le \left\lfloor \frac{T_j}{k-\ell} \right\rfloor, \ \forall k=0,\dots,\Gamma_j,\ \ell=0,\dots,M_j,\ k>\ell,\ j=1,\dots,6, \\
& s^2_{k_1\ell_2\ell_4}=0,\ \forall k_1>\ell_2,\ \forall \ell_4\ \ s^2_{k_1\ell_2\ell_4}=0,\ \forall k_1>\ell_4,\ \forall \ell_2,\ s^5_{k_2k_4\ell_5}=0,\ \forall k_4>\ell_5, \forall k_2, \\
& s^6_{k_2\ell_3\ell_5}=0,\ \forall k_2>\ell_3, \forall \ell_5,\ \ s^9_{\ell_6k_3k_5}=0,\ \forall k_3>\ell_6, \forall k_5,\ \ s^9_{\ell_6k_3k_5}=0,\ \forall k_5>\ell_6,\ \forall k_3, \\
& r^2_{k_2\ell_2}=\sum_{k_4} s^4_{\ell_2k_2k_4}, \forall k_2,\ell_2,\ \ r^3_{k_3\ell_3}=\sum_{k_5} s^8_{\ell_3k_3k_5}, \forall k_3,\ell_3,\ \ r^4_{k_4\ell_4}=\sum_{\ell_2} s^3_{\ell_2\ell_4k_4}, \forall \ell_4,k_4, \\
& r^5_{k_5\ell_5}=\sum_{\ell_3} s^7_{\ell_3\ell_5k_5}, \forall \ell_5,k_5,\ \ \sum_{\ell_1} r^1_{k_1\ell_1}=\sum_{\ell_2}\sum_{\ell_4} s^2_{k_1\ell_2\ell_4}, \forall k_1,\ \ \sum_{k_6} r^6_{k_6\ell_6}=\sum_{k_3}\sum_{k_6} s^9_{\ell_6k_3k_5}, \forall \ell_6, \\
& \sum_{k_1} s^2_{k_1\ell_2\ell_4}=\sum_{k_4} s^3_{\ell_2\ell_4k_4}, \forall \ell_2,\ell_4,\ \ \sum_{\ell_4} s^3_{\ell_2\ell_4k_4}=\sum_{k_2} s^4_{\ell_2k_2k_4}, \forall \ell_2,k_4,\ \ \sum_{\ell_2} s^4_{\ell_2k_2k_4}=\sum_{\ell_5} s^5_{k_2k_4\ell_5}, \forall k_2,k_4, \\
& \sum_{k_4} s^5_{k_2k_4\ell_5}=\sum_{\ell_3} s^6_{k_2\ell_3\ell_5}, \forall k_2,\ell_5,\ \ \sum_{k_2} s^6_{k_2\ell_3\ell_5}=\sum_{k_5} s^7_{\ell_3\ell_5k_5}, \forall \ell_3,\ell_5,\ \ \sum_{\ell_5} s^7_{\ell_3\ell_5k_5}=\sum_{k_3} s^8_{\ell_3k_3k_5}, \forall \ell_3,k_5, \\
& \sum_{\ell_3} s^8_{\ell_3k_3k_5}=\sum_{\ell_6} s^9_{\ell_6k_3k_5}, \forall k_3,k_5,\ \ \{r^1_{k_1\ell_1}\},\{s^2_{k_1\ell_2\ell_4}\},\dots,\{s^9_{\ell_6k_3k_5}\},\{r^6_{k_6,\ell_6}\}\in\Delta.
\end{aligned}
$$

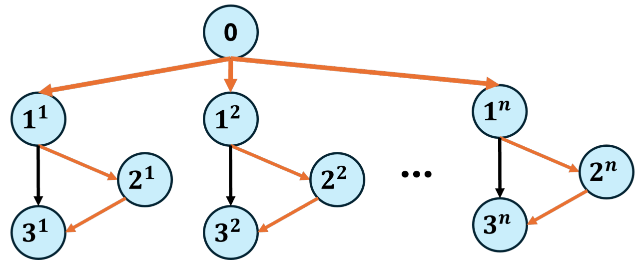
E.3. Illustrative example for Section 5.2

EXAMPLE EC.1. Consider the graph G given in Figure EC.7 with $3n+1$ nodes and the following parameters: $h_{1^k}=h_{3^k}=h_0=2$, $h_{2^k}=1$, $T_{1^k}=T_{2^k}=1$, $T_{3^k}=T_0=0$, $S_{3^k}\le 0$, $\forall k=1,\dots,n$. Furthermore, let $SI_0=0$, $\sigma_0=1$, and $\sigma_{i^k}=1$ for all $i\in\{1,2,3\}$ and $k=1,\dots,n$.

We wish to solve the GSM:

$$
\min \qquad h_0\sigma_0\sqrt{SI_0+T_0-S_0}+\sum_{k=1}^{n}\sum_{i=1}^{3} h_{i^k}\sigma_{i^k}\sqrt{SI_{i^k}+T_{i^k}-S_{i^k}}
$$

Figure EC.7 An example where our approach outperforms that in Humair and Willems (2011).



$$
\begin{aligned}
\text{s.t.} \quad & SI_{i^k} + T_{i^k} - S_{i^k} \geq 0, \ \forall i = 1, \ldots, 3, \ \forall k = 1 \ldots, n, \\
& SI_0 + T_0 - S_0 \geq 0, \\
& S_0 \leq SI_{1^k}, S_{1^k} \leq SI_{2^k}, S_{2^k} \leq SI_{3^k}, S_{1_k} \leq SI_{3^k}, S_{3^k} \leq 0, \ \forall k, SI_0 = 0 \\
& 0 \leq S_0, SI_0, S_{i^k}, SI_{i^k} \leq 2, \ \forall i, k.
\end{aligned}
$$

As $tw(G) = 2$ and $M = 2$, the LP we propose in Section 4 to solve the problem has $O(n)$ constraints and variables.

We take the root tree to be the tree spanned by the orange edges in Figure EC.7. This is not the tree suggested by Humair and Willems (2011, EC5.1), which leads to an even larger number of DPs being solved. Hence, the tree indicated in Figure EC.7 provides a conservative estimate for solution time of the DP.

On the chosen tree, the solution approach proposed in Humair and Willems (2011) requires solving $2^{n+1} - 1$ dynamic programs over the tree. Indeed, one can see that the optimal solution of the GSM on the root tree places 2 units of safety stock at each node 2^k, where holding costs are cheapest. This violates the constraints $S_{1^k} \leq SI_{3^k}$ for all $k = 1, \ldots, n$. Resolving the violated constraint in each triangle has no impact on the other triangles as node 0 effectively decouples the solutions from one another. Thus, we need to branch on all violated edges: this leads to a tree with 2^n leaves, where only the last visited leaf (per the algorithm) contains the optimal solution. As a dynamic program is solved in each node of the tree, it follows that $2^{n+1} - 1$ dynamic programs are being solved.

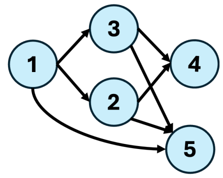
This example illustrates a setting where our approach significantly outperforms Humair and Willems (2011): our LP scales linearly with the dimension n whereas the branch-and-bound approach scales exponentially with n.

Appendix F: Proofs and additional results relating to Section 6

F.1. Deriving principled lower bounds on (2)

Proof of Proposition 4. For the first claim, it is enough to show that any solution to (12) can be converted into a solution to (4) with the same objective value. Let $\{r^j_{k\ell}\}$ and $\{s^z_{K_z,L_z}\}$ be a solution to (12). Letting $\mathcal{T} = (T, \{X_z\}_{z \in T})$ be a tree decomposition of G', for given $(i, j) \in E$, there must exist a node $z_{ij} \in T$ such that $X_{z_{ij}}$ contains S_i and SI_j. Define $q^{ij}_{k_i \ell_j} = \sum_{K_{z_{ij}} \setminus \{k_i\}} \sum_{L_{z_{ij}} \setminus \{\ell_j\}} s^{z_{ij}}_{K_{z_{ij}} L_{z_{ij}}} \ \forall k_i, \ell_j$. It can then be shown that $\{r^j_{k\ell}\}$ and $\{q^{ij}_{k\ell}\}$ as defined are solutions to (4). Indeed, ($12a_j$) straightforwardly implies that ($4a_j$), ($12b^z_{i_z j_z}$) implies ($4b_{ij}$) (as the sum of zeros is zero), ($4c_j$) and ($4d_i$) follow from ($12c^z_{j_z}$) and ($12d^z$), and finally ($4e_j$) and ($4f_{ij}$) are a consequence of ($12e^z$). Equality of objective values is immediate. The second claim follows from the fact that (4) is equivalent to (2) if its decision variables are binary (see Proposition EC.1 in Appendix C) and (12) is equivalent to (2) (see Theorem 2). □

Figure EC.8 Graph G used to illustrate the hierarchy of lower bounds.



Consider the graph G on 5 nodes in Figure EC.8, obtained by adding node 5 to the diamond structure in Figure 2. We reprise the same values for T_j and $h_j\sigma_j$ for $j = 1, 2, 3, 4$, as in Section 4.1, with $d_4 = 1$. For node 5, we take $T_5 = 0$ and $h_5\sigma_5 = 0.01$, i.e., node 5 is a low-cost node, and we let $d_5 = 0$. The intersection graph G' of (2) on G is given in Figure EC.9. Its treewidth is 3 and its tree decomposition is given in Figure EC.10. As a consequence, writing the full linear program (12) would require variables keeping track of four service-times jointly and would thus be of size $O(nM^4)$, where M is the maximum replenishment time.

We show that, for this example, (i) the LP (4), which tracks pairs of variables and is of size $O(nM^2)$, yields a strict lower bound on (2); and (ii) there exists an LP which tracks *triples* of variables jointly, is of size $O(nM^3)$, and yields an optimal solution to (2). Thus, one need not immediately consider the full LP (12) when the lower bound (4) fails to be tight, as there may exist LP formulations of intermediate size which yield the optimal solution already.

Figure EC.9 The intersection graph G' for G as given in Figure EC.8.

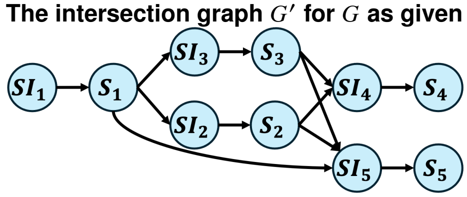
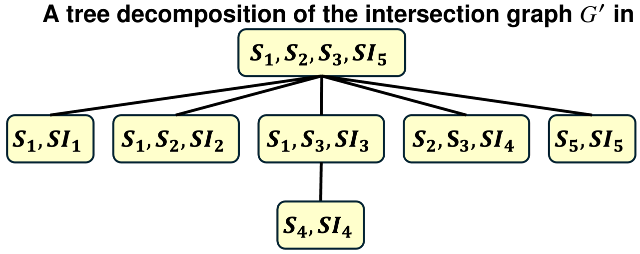


Figure EC.10 A tree decomposition of the intersection graph G' in Figure EC.9.



Solving (2), or equivalently (12), on G. It is not difficult to see that the optimal solution to this problem is given by $\{S_j\}_{j=1}^5 = (1, 2, 1, 1, 0)$ and $\{SI_j\}_{j=1}^5 = (0, 1, 1, 2, 2)$ with an optimal value of $2\sqrt{2} + 0.01\sqrt{2} \approx 2.843$. Indeed, without node 5, the optimal solution and value is given in Section 4.1. When we add node 5, we add the constraint $SI_5 \geq \max\{S_1, S_2, S_3\}$. At the optimal solution from Section 4.1, this requires $SI_5 \geq 2$, while $S_5 = 0$ because of $d_5 = 0$. Given the small cost of node 5, it remains optimal to keep the diamond solution unchanged and set $SI_5 = 2$ and $S_5 = 0$, thereby incurring only the additional cost $0.01\sqrt{2}$.

Solving (4) on G. Similarly to Section 4.1, the LP (4) fails to obtain an exact solution to (2). Indeed, a standard solver returns the optimal fractional solution $r^1_{00} = r^1_{10} = r^2_{10} = r^2_{01} = r^3_{00} = r^3_{11} = r^4_{10} = r^4_{11} = r^5_{00} = r^5_{01} = \frac{1}{2}$ and $q^{ij}_{00} = q^{ij}_{11} = \frac{1}{2}$ for every $(i, j) \in E$, with all remaining $r^j_{k\ell} = q^{ij}_{k\ell} = 0$. The optimal value is then $2 + \sqrt{2}/2 + 0.01/2 \approx 2.712 < 2\sqrt{2} + 0.01\sqrt{2}$, which showcases that (4) provides a strict lower bound on the true optimal value of the GSM.

An LP formulation between (4) and (12) that solves the GSM to optimality. The key idea is to introduce the variables and constraints corresponding to (12) for the diamond subgraph 1, 2, 3, 4, while maintaining the variables and constraints of (4) associated to node 5. More specifically, let

$C_1 = \{SI_1, S_1\}$, $C_2 = \{S_1, SI_2, SI_3\}$, $C_3 = \{SI_3, SI_2, S_3\}$, $C_4 = \{SI_2, S_2, S_3\}$, $C_5 = \{S_3, S_2, SI_4\}$, $C_6 = \{SI_4, S_4\}$,

where we associate each cluster to a corresponding family of variables:

$$\{s^1_{\ell_1 k_1}\}, \quad \{s^2_{k_1\ell_2\ell_3}\}, \quad \{s^3_{\ell_3\ell_2k_3}\}, \quad \{s^4_{\ell_2k_2k_3}\}, \quad \{s^5_{k_3k_2\ell_4}\}, \quad \{s^6_{\ell_4k_4}\}.$$

We also work with the node variables $r^j_{k_j\ell_j}$, $j = 1, \dots, 5$, and edge variables $q^{i5}_{k_i\ell_5}$, $i = 1, 2, 3$, for the arcs entering node 5. The LP of interest is then given by

$$\begin{aligned}
\min \quad & \sum_{j=1}^{5}\sum_{k_j=0}^{\Gamma_j}\sum_{\ell_j=0}^{M_j} h_j\sigma_j\sqrt{\ell_j+T_j-k_j}\, r^j_{k_j\ell_j} \qquad \text{(EC.9)}\\
\text{s.t.} \quad & r^j_{k_j\ell_j} \le \left\lfloor \frac{T_j}{k_j-\ell_j}\right\rfloor, \ \forall j\in V,\ \forall k_j>\ell_j,\\
& q^{i5}_{k_i\ell_5}=0,\ \forall i\in\{1,2,3\},\ \forall k_i>\ell_5, \quad s^2_{k_1\ell_2\ell_3}=0,\ \forall k_1>\ell_2 \text{ or } k_1>\ell_3, \quad s^5_{k_3k_2\ell_4}=0,\ \forall k_2>\ell_4 \text{ or } k_3>\ell_4,\\
& r^1_{k_1\ell_1}=s^1_{\ell_1k_1},\ \forall k_1,\ell_1, \quad r^2_{k_2\ell_2}=\sum_{k_3}s^4_{\ell_2k_2k_3},\ \forall k_2,\ell_2, \quad r^3_{k_3\ell_3}=\sum_{\ell_2}s^3_{\ell_3\ell_2k_3},\ \forall k_3,\ell_3, \quad r^4_{k_4\ell_4}=s^6_{\ell_4k_4}, \forall k_4,\ell_4,\\
& \sum_{\ell_1}s^1_{\ell_1k_1}=\sum_{\ell_2,\ell_3}s^2_{k_1\ell_2\ell_3},\ \forall k_1, \quad \sum_{k_1}s^2_{k_1\ell_2\ell_3}=\sum_{k_3}s^3_{\ell_3\ell_2k_3},\ \forall \ell_2,\ell_3, \quad \sum_{\ell_3}s^3_{\ell_3\ell_2k_3}=\sum_{k_2}s^4_{\ell_2k_2k_3},\ \forall \ell_2,k_3,\\
& \sum_{\ell_2}s^4_{\ell_2k_2k_3}=\sum_{\ell_4}s^5_{k_3k_2\ell_4},\ \forall k_2,k_3, \quad \sum_{k_3,k_2}s^5_{k_3k_2\ell_4}=\sum_{k_4}s^6_{\ell_4k_4},\ \forall \ell_4, \quad \sum_{\ell_1}s^1_{\ell_1k_1}=\sum_{\ell_5}q^{15}_{k_1\ell_5},\ \forall k_1,\\
& \sum_{\ell_2,k_3}s^4_{\ell_2k_2k_3}=\sum_{\ell_5}q^{25}_{k_2\ell_5},\ \forall k_2, \quad \sum_{\ell_3,\ell_2}s^3_{\ell_3\ell_2k_3}=\sum_{\ell_5}q^{35}_{k_3\ell_5},\ \forall k_3, \quad \sum_{k_i}q^{i5}_{k_i\ell_5}=\sum_{k_5}r^5_{k_5\ell_5}, \quad \forall i\in\{1,2,3\},\ \forall \ell_5,\\
& \{s^1_{\ell_1k_1}\}_{\ell_1,k_1}, \{s^2_{k_1\ell_2\ell_3}\}_{k_1,\ell_2,\ell_3}, \{s^3_{\ell_3\ell_2k_3}\}_{\ell_3,\ell_2,k_3}, \{s^4_{\ell_2k_2k_3}\}_{\ell_2,k_2,k_3}, \{s^5_{k_3k_2\ell_4}\}_{k_3,k_2,\ell_4}, \{s^6_{\ell_4k_4}\}_{\ell_4,k_4}\in\Delta,\\
& \{r^5_{k_5\ell_5}\}_{k_5,\ell_5}\in\Delta, \qquad \{q^{i5}_{k_i\ell_5}\}_{k_i,\ell_5}\in\Delta, \quad i=1,2,3.
\end{aligned}$$

The optimal value of (EC.9) is a valid lower bound on (12), and hence on (2). Moreover, this lower bound is at least as large as the one obtained from (4), since (EC.9) imposes additional consistency requirements on the diamond subgraph. Solving (EC.9) yields a binary optimal solution with value $2\sqrt{2}+0.01\sqrt{2}$, which matches the true optimal value of the GSM. Hence, for this instance, the intermediate LP closes the gap left by (4) without requiring the full LP (12).

A hierarchy of LPs of increasing size. In this example, the three LPs can be viewed as successive levels of a hierarchy indexed by the largest marginal block size minus one. The pairwise LP (4) corresponds to $k = 1$, since it tracks pairs of variables. The intermediate formulation (EC.9) corresponds to $k = 2$, since it adds selected triple marginals on the diamond. The full formulation (12) corresponds to $k = 3$, since the tree decomposition of the intersection graph contains a bag with four variables. Let OPT_k denote the optimal value of the corresponding formulation. Then, $OPT_1 \le OPT_2 \le OPT_3$, where OPT_3 equals the optimal value of the GSM by Theorem 2. In this example, the strict inequality $OPT_1 < OPT_3$ shows that the pairwise relaxation is not tight, while the equality $OPT_2 = OPT_3$ shows that adding triple marginals already leads to an exact solution. Thus, when (4) fails to be tight, one need not necessarily jump immediately to the full treewidth LP. LPs of intermediate size may already close the gap.

Here, the triple clusters were chosen using the intuition from Section 4.1: (4) fails because of inconsistent mixing across the two paths of the diamond and may be resolved by adding variables tracking the clusters C_1 through C_6. Developing general, principled rules for selecting such intermediate clusters is an interesting direction for future work.

F.2. Deriving upper bounds on (2) in a principled way

To obtain upper bounds on (2), the possible values of S_j and SI_j in the LP can be coarsened. Until now, we have included *all* integer values between 0 and Γ_j (for S_j) and 0 and M_j (for SI_j). In practice, one may restrict these variables to a smaller subset of integer values, thereby reducing the LP size (and thus solving time), but potentially increasing the objective value (since the LP becomes more restrictive). We now provide the exact formulation of this LP. Let $\alpha \in \mathbb{N}_{>0}$ be a coarsening factor, and define

$$\tilde{\mathcal{D}}_{K_z} := \left\{0,\ldots,\left\lfloor\frac{\Gamma_{i_1}}{\alpha}\right\rfloor\right\}\times\ldots\times\left\{0,\ldots,\left\lfloor\frac{\Gamma_{i_{|I_z|}}}{\alpha}\right\rfloor\right\} \text{ and } \tilde{\mathcal{D}}_{L_z} := \left\{0,\ldots,\left\lfloor\frac{M_{j_1}}{\alpha}\right\rfloor\right\}\times\ldots\times\left\{0,\ldots,\left\lfloor\frac{M_{j_{|J_z|}}}{\alpha}\right\rfloor\right\}.$$

Then, let

$$\tilde{\mathcal{S}}_{i_z j_z} = \left\{((K_z, L_z) \,\middle|\, K_z \in \tilde{\mathcal{D}}_{K_z}, L_z \in \tilde{\mathcal{D}}_{L_z},\ k_{i_z} > \ell_{j_z}\right\},\ \forall (i_z, j_z) \in (I_z \times J_z) \cap E, z \in T.$$

We then construct the following LP, which provides an upper bound for (12):

$$\min \quad \sum_{j\in V}\sum_{\ell=0}^{\left\lfloor\frac{M_j}{\alpha}\right\rfloor}\sum_{k=0}^{\left\lfloor\frac{\Gamma_j}{\alpha}\right\rfloor} f_j\left(\alpha(\ell-k)\right) r^j_{k\ell} \tag{$\tilde{12}$}$$

$$\text{s.t.} \quad r^j_{k\ell} \le \left\lfloor\frac{\left\lfloor\frac{T_j}{\alpha}\right\rfloor}{k-\ell}\right\rfloor,\ \forall k=0,\ldots,\left\lfloor\frac{\Gamma_j}{\alpha}\right\rfloor,\ \ell=0,\ldots,\left\lfloor\frac{M_j}{\alpha}\right\rfloor,\ k>\ell,\ j\in V, \tag{$\tilde{12}a^j$}$$

$$s^z_{K_z L_z} = 0,\ \forall (K_z, L_z) \in \tilde{\mathcal{S}}_{i_z j_z},\ \forall (i_z, j_z) \in (I_z, J_z) \cap E,\ z \in T, \tag{$\tilde{12}b^z_{i_z j_z}$}$$

$$(\tilde{12}c^z_{j_z}), (\tilde{12}d^z_{j_z}), (\tilde{12}e^z_{i_z}), (\tilde{12}f^z),$$

where ($\tilde{12}x$) equals ($12x$) but with appropriately adjusted indices. If $\alpha = 1$, then ($\tilde{12}$) is (12). One could also envision a sequential procedure, conceptually similar to branch-and-bound, where we solve a coarse LP, then refine the grid of (S_j, SI_j) values around the current solution, and re-solve as needed until the coarseness is acceptable. This yields a sequence of improving upper bounds.

Numerical experiments. We next describe numerical experiments on synthetic data highlighting the upper bounds. Before, we briefly discuss how we generate our data.

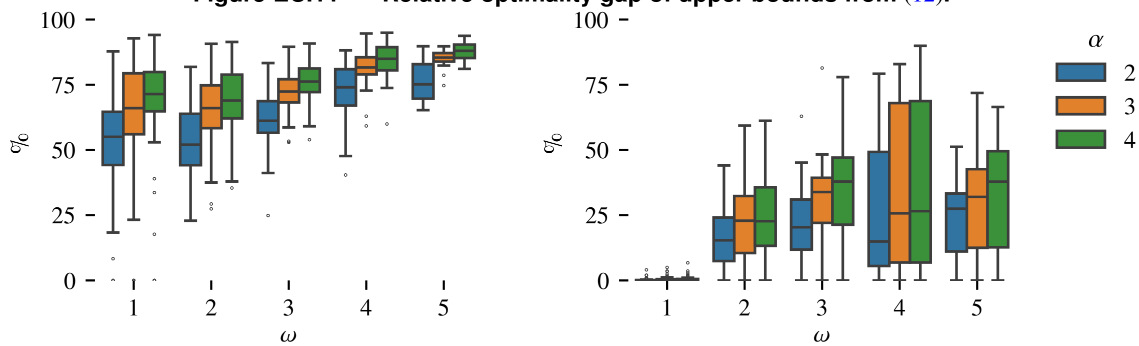
Network generation. Tiered supply chain networks G are generated using the random generative model Blaettchen et al. (2024) with $n \in \{20, 40, 60, 80\}$ nodes, $k \in \{2, 3, 4, 5\}$ tiers, connectivity parameters $c \in \{1, 2, 3, 4, 5\}$, and $p \in \{0.25, 0.5, 0.75, 1.0\}$. Nodes are distributed across tiers $\ell = 1, \ldots, k$, where k is the most downstream tier, with a higher probability of nodes being assigned to upstream tiers ("converging") or to downstream tiers ("diverging"). That is, we vary the vector η that controls the probability of a node being assigned to a tier,[6] by generating two structural archetypes: (i) "converging" with $\eta_\ell = 0.1 + 2\frac{1-0.1k}{k(k-1)}(k-\ell)$, and (ii) "diverging" with $\eta_\ell = 0.1 + 2\frac{1-0.1k}{k(k-1)}(\ell - 1)$.

GSM parameters. First, we assign a random processing time $T_j \in \{1, \ldots, 4\}$ for each nodes j and a holding cost $h_j = 3$ for all nodes j at tier k. Then, the inventory holding cost for each subsequent tier is increased by a factor $\tilde{h} = \{1.0, 1.3, 1.6\}$. We also set the service bounds d_j to zero for all sink nodes, that is, all nodes at tier 1.

Second, we assume $D_j(\tau) = \mu_j\tau + \sigma_j\sqrt{\tau}$ for each node j. Standard deviations at sink nodes L are randomly drawn from $\{5, \ldots, 10\}$. Then, for tiers $\ell = 2, \ldots, k$, we set $\sigma_i = \sqrt{\sum_{j\,:\,j\in L}\left(|paths(i,j)|\sigma_j\right)^2}$, where $|paths(i,j)|$ indicates the number of distinct paths from i to j.

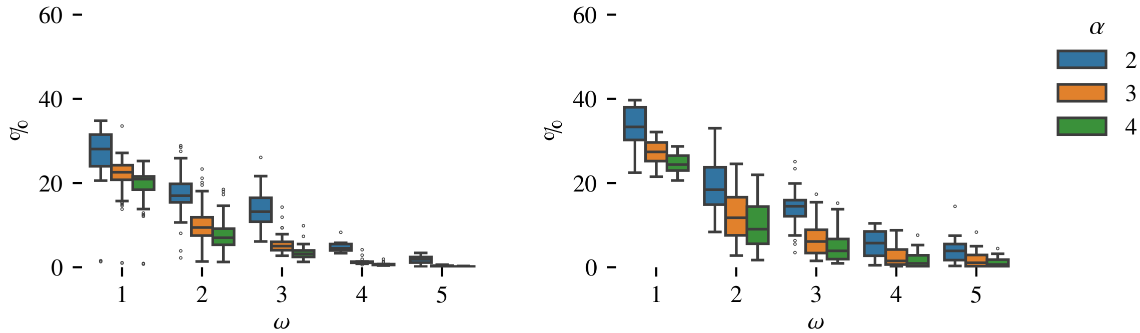
[6] This vector is termed α in Blaettchen et al. (2024), but we change its notation to avoid conflicts.

Figure EC.11 Relative optimality gap of upper bounds from ($\widetilde{12}$).



(a) Converging structure. (b) Diverging structure.

Figure EC.12 Percentage of time taken to solve ($\widetilde{12}$) relative to (12).



(a) Converging structure. (b) Diverging structure.

Instances and solving. For each combination of possible options, we generate two instances. We obtain (upper bounds on) the treewidths using the highly efficient algorithm presented in Hamann and Strasser (2018). After sorting out networks where the intersection graph's treewidth exceeds 5, or where the building time for LP (12) or ($\widetilde{12}$) exceeds 5 minutes with an Apple M4 Pro processor and 48 GB RAM of memory, we retain 1,106 instances. Problems (12) ($\alpha = 1$) and ($\widetilde{12}$) ($\alpha \in \{2, 3, 4\}$) are then solved using Gurobi. Note that Figures EC.11 and EC.12 only display the results for instances with GSM parameter $\tilde{h} = 1.3$ for clarity. The results for other values of $\tilde{h}$ are qualitatively unchanged.

Main findings. Figures EC.11 and EC.12 summarize numerical experiments where we apply the coarsening approach to supply chain instances generated by Blaettchen et al. (2024). Figure EC.11 shows how the bound quality improves as the coarsening factor α decreases and also demonstrates that converging and diverging supply chains exhibit different behaviors: the upper bounds improve more quickly for diverging networks (especially with smaller treewidth). Figure EC.12 reports the corresponding computation times as a percentage of the time required by the full LP (12).

F.3. Computational experiments for Section 6.2

This section provides complete details on the computational experiments summarized in Section 6.2. We describe the experimental setup, benchmark instances, instance characteristics, and detailed results.

Experimental setup. We cap the run time of all algorithms at two hours. All instances were solved on compute nodes with an Intel Xeon Gold 6226 CPU processor with 2.7 GHz clock speed and 48 GB RAM. To solve the LP, we use Gurobi 12.0 and Pyomo as the modeling language. For BARON, we use version 2025.03.19, also with Pyomo.

We use our own implementation of the DP and the heuristic HGNA in Humair and Willems (2011), as the code from the original paper is not available. Our initial lower bounds on S_j and SI_j are 0 and our initial upper bounds are obtained via the longest stage-time path through each stage (see page 5 of the paper's electronic companion). The spanning tree, initial upper bound on the optimal value, and the sequence in which infeasible links are picked, all follow from the best practices recommended in the paper's electronic companion.

For the LP, we can obtain lower bounds on the optimal value of the GSM by solving (4) instead of (12) (see Proposition 4). For each instance, we check whether the solution to (4) is binary. It turns out that, in all but one instance (instance 28), the solution is binary, which implies that LP (4) returns an optimal solution to (12) (see Proposition 4 again). For instance 28, the solution produced by (4) is non-binary, but adding a very small linear tie-breaking perturbation term to the objective is sufficient to recover the optimal binary solution. We note that similar results are observed on randomly generated supply chain network instances, indicating that the lower bound provided by (4) is extremely powerful. We report both the set-up time of the LP and its solving time.

Benchmark instances and problem parameters. We consider the 38 supply chain networks published by Willems (2008). For each network, the dataset provides the graph $G = (V, E)$, the processing times $T_j\ \forall j \in V$, and the direct cost added at a processing stage, $c_j\ \forall j \in V$. For demand nodes $j \in L$, averages μ_j and standard deviations σ_j of periodic demands are also provided, as well as (percentage) service levels β_j, and maximum services times d_j.

To compute the cost function, we require both an expression of the safety stock as a function of service times, and of the per-unit holding costs. For the former, we start at the demand nodes. In particular, we assume a normal distribution of demand, and let $\overline{d}_j = \mu_j + \Phi^{-1}(\beta_j) \cdot \sigma_j\ \forall j \in L$, where Φ^{-1} is the inverse of the cumulative distribution function of the Standard Normal distribution. We then propagate demand upwards in the network. That is, for a node $i \in V \setminus L$, we let $\overline{d}_i = \sum_{j\in L} \sum_{p\in paths(G)\,:\, i,j\in p} \overline{d}_j$ and $\mu_i = \sum_{j\in L} \sum_{p\in paths(G)\,:\, i,j\in p} \mu_j$. Finally, we assume that the safety stock required at node $j \in V$ is $(\overline{d}_j - \mu_j)\sqrt{SI_j + T_j - S_j}$. As the cost function used in Humair and Willems (2011) is not fully specified, we choose this simplified demand function to avoid assumptions about pooling.

Holding costs are commonly computed as a percentage of the total value of the item currently in inventory. Hence, to obtain the holding costs h_j at node $j \in V$, we take the total direct cost added up to and including node j. That is, we first let $h_j = c_j$ for all nodes j without predecessors. Denote the set of such nodes as V_0. Next, we iteratively build the sets of nodes V_ℓ for $\ell = 1, 2, \ldots$. In particular, V_ℓ is defined as the set of nodes in $V \setminus (V_0 \cup \ldots \cup V_{\ell-1})$ whose predecessors are all in $V_0 \cup \ldots \cup V_{\ell-1}$. We then let $h_j = c_j + \sum_{i\in V\,:\,(i,j)\in E} h_i\ \forall j \in V_\ell$. Finally, the cost function for $j \in V$ is $f_j(SI_j - S_j) = h_j(\overline{d}_j - \mu_j)\sqrt{SI_j + T_j - S_j}$.

We emphasize that the same parameters and function definitions are used under all solving procedures. Alternative definitions of the cost function, in particular, do not affect our results.

Instance characteristics and runtime performance. Table EC.3 summarizes instance characteristics and solver runtimes for all 38 benchmark instances from Willems (2008). In the left hand columns, $|V|$ and $|E|$ denote the number of nodes and arcs in the supply chain network G, while ω' is the treewidth of the intersection graph G'. The columns $\max \Gamma_j$ and $\max M_j$ report the largest upper bounds on outbound service time S_j and inbound service time SI_j across all nodes in an instance. In the right hand columns, we report solving times for each method: our LP approach in the *LP* column; BARON used on (2) in the *MINLP* column; and the DP approach from Humair and Willems (2011) in the *GNA* column. We emphasize that, as explained in Section 6.2, we do not in fact need to solve LP (12) to solve the GSM (2), as the much smaller LP (4) returns an optimal solution to (2) every time.

Table EC.3 Instance characteristics and solver runtime performance

	Instance characteristics					*Runtime performance*		
Inst.	$\lvert V\rvert$	$\lvert E\rvert$	ω'	$\max_j \Gamma_j$	$\max_j M_j$	LP (s)	MINLP (s)	GNA (s)
01	8	10	2	38	38	0.2	2.6	0.0
02	13	13	2	59	64	1.2	6.2	0.4
03	17	18	2	77	79	1.6	26	1.2
04	22	39	4	190	204	14	123	7,200†
05	27	31	3	42	47	1.2	18	0.3
06	28	28	2	82	96	2.7	179	0.1
07	38	78	6	78	85	4.3	7,200†	79
08	40	48	2	86	91	4.0	7,200†	16
09	49	52	3	41	47	1.5	0.3	0.1
10	58	176	6	122	162	25	2.2	2.9
11	68	108	6	55	60	3.9	7,200†	7,200†
12	88	107	5	85	108	9.4	7,200†	116
13	108	452	8	20	26	4.1	3,265	7,200†
14	116	119	2	131	131	18	7,200†	0.6
15	133	164	3	26	26	1.7	7,200†	0.1
16	145	224	8	158	163	66	7,200†	7,201†
17	152	211	3	47	57	7.1	7,200†	0.2
18	154	224	6	100	100	18	7,200†	217
19	156	263	6	116	125	25	7,200†	7,200†
20	156	169	2	159	160	12	7,200†	7,200†
21	186	359	14	91	96	22	7,200†	7,200†
22	253	253	2	569	691	1,715	7,200†	38
23	271	524	20	77	77	15	7,200†	7,200†
24	334	1,245	19	64	68	63	7,200†	7,200†
25	409	853	21	54	82	64	7,200†	7,201†
26	468	605	3	384	394	528	7,200†	313
27	482	941	13	100	105	155	7,200†	7,201†
28	577	2,262	24	113	123	1,055	7,200†	7,201†
29	617	753	11	31	43	7.2	7,200†	10
30	626	632	2	71	71	13	7,200†	464
31	706	908	12	12	17	3.4	7,200†	672
32	844	1,685	24	108	112	135	7,200†	7,201†
33	976	1,009	4	72	72	49	7,200†	4.4
34	1,206	4,063	35	86	89	380	7,200†	7,201†
35	1,386	1,857	8	77	81	64	7,201†	7,201†
36	1,451	4,812	45	47	49	75	7,202†	7,201†
37	1,479	2,069	24	16	27	13	7,200†	7,201†
38	2,025	16,225	87	17	26	433	7,201†	7,201†
Min	8	10	2	12	17	0.2	0.3	0.0
Max	2,025	16,225	87	569	691	1,715	7,202†	7,201†
Median	156	258	6	77	82	14	7,200†	568

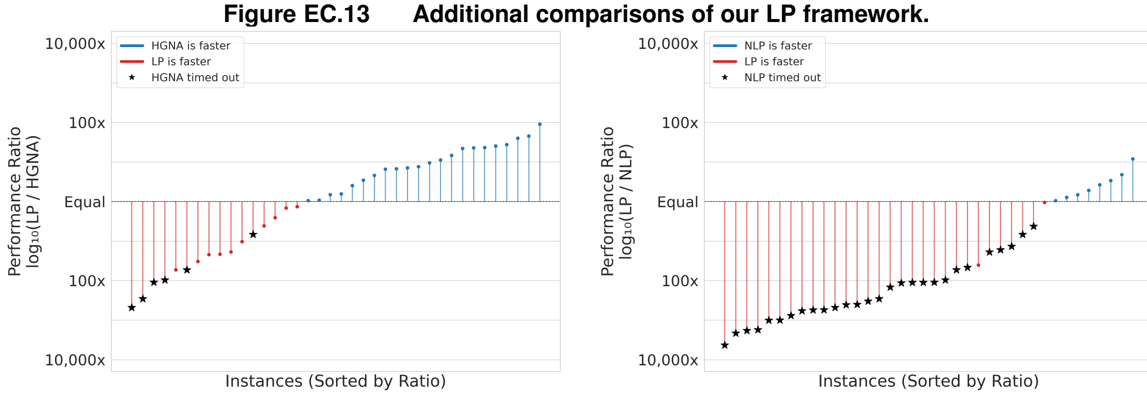
Note. ω' = treewidth of intersection graph; LP = total LP time (build + solve); MINLP = BARON with integer service times; GNA = Humair-Willems General Networks Algorithm. † denotes 2-hour timeout reached.

Large-instance performance. For the 10 largest instances ($\lvert V\rvert \geq 617$), the lower-bound LP in (4) solves all instances to optimality in under 433 seconds, with a median solve time of 57 seconds. In contrast, BARON times out on all 10 instances, and GNA times out on 6 of the instances. This demonstrates that the LP is the only method that reliably scales to large supply chain networks.

Comparison with HGNA heuristic. Besides their exact algorithm, Humair and Willems (2011) also propose a heuristic, HGNA, which we compare to our LP-based framework. Note that HGNA does *not* provably recover an optimal solution, whereas our LP does. The log-ratios of the total time to solve the LP to the time taken for HGNA are given in Figure EC.13a. We observe that HGNA times out on 6 of the 38 instances, which the LP does not. Furthermore, the LP improves on the solving time of HGNA on 16 out of the 38 instances, i.e., 42%. Thus, even the heuristic in Humair and Willems (2011), which was designed with the goal of speeding up computation (at the expense of optimality guarantees) is only moderately competitive against our LP, which maintains optimality guarantees. We remark that these 16 instances correspond to the larger instances in the dataset.

Comparison with NLP formulation. As explained in Section 6.2, we also used BARON to solve (2) without integrality constraints (NLP mode). The log-ratios are given in Figure EC.13b. The NLP formulation and MINLP formulation are theoretically equivalent on the instances in Willems (2008), as T_j and d_j are integers for all $j \in V$. In practice, the speed-ups obtained by dropping integrality are modest: the NLP's mean solve time is 5,312s, its median solve time remains at the 2-hour timeout, and the LP remains faster than the NLP in 30 of the 38 instances. This indicates that the computational bottleneck for commercial solvers is the non-convex objective of (2) rather than the integrality constraints. Switching from MINLP to NLP does not yield sufficient gains to become competitive with the DP formulation, GNA, either.

Figure EC.13 Additional comparisons of our LP framework.



(a) Comparison with the HGNA heuristic in Humair and Willems (2011).

(b) Comparison with BARON in NLP mode.

F.4. Proofs of results for the sensitivity analysis in Section 6.3

Proof of Proposition 5.

Proof. Let $b_i \in D$. Recall that (y_{b_i}, v_{b_i}) are the dual variables of (LP_{b_i}). Note that (y_{b_i}, v_{b_i}) is also feasible to the dual of (LP_b). Thus, by weak duality, for any $\bar{b} \in D$, $\Phi(\bar{b}) \geq y_{b_i}^T \bar{b} + h^T v_{b_i}$. The result follows by considering a set of values $\{b_1, \ldots, b_r\}$ and taking the largest lower bound. □

Procedure to leverage Proposition 5 for $\{d_j\}_{j\in L}$. Parameters $\{d_j\}_{j\in L}$ do not explicitly feature as right-hand-side constraints in (4), making the application of Proposition 5 challenging. Hence, we present a reformulation of (4).

Let $\{d_j^1\},\dots,\{d_j^r\}_{j\in L}$ be upper bounds on service times at external demand nodes L. These play the role of $\{b_1,\dots,b_r\}$ in Proposition 5, i.e., we assume that we are able to solve a linear program (defined below) with the service times fixed and extract dual variables. We further let $\{\hat{d}_j\}_{j\in L}$ be the upper bounds on service times at L that play the role of $\bar{b}$ in Proposition 5, i.e., we aim to obtain lower bounds on the holding cost associated with such service times, without solving any new linear program. Let $\tilde{\Gamma}_j=\min\{\max\{d_j^1,\dots,d_j^r,\hat{d}_j\},M_j\}$ for all $j\in L$ and $\tilde{\Gamma}_j=\Gamma_j$ for $j\notin L$.

PROPOSITION EC.10. *For $\zeta\in\{1,\dots,r\}$, solving (4) for external customer demands $\{d_j^\zeta\}_{j\in L}$ is equivalent to:*

$$
\begin{aligned}
\min\quad & \sum_{j\in V}\sum_{\ell=0}^{M_j}\sum_{k=0}^{\tilde{\Gamma}_j} f_j(\ell-k)r_{k\ell}^j && \text{(EC.10)}\\
\text{s.t.}\quad & r_{k\ell}^j\le\left\lfloor\frac{T_j}{k-\ell}\right\rfloor,\ \forall k=0,\dots,\tilde{\Gamma}_j,\ \ell=0,\dots,M,\ k>\ell,\ j\in V,\\
& q_{k_i\ell}^{ij}=0,\ \forall(k_i,\ell)\in\mathcal{S}_{ij},\ \forall(i,j)\in E,\\
& \sum_{k=0}^{\tilde{\Gamma}_j} r_{k\ell}^j=\sum_{k_i=0}^{\tilde{\Gamma}_i} q_{k_i\ell}^{ij},\ \forall\ell=0,\dots,M_j,\ \forall j\in V\ \text{s.t. } (i,j)\in E,\\
& \sum_{\ell_i=0}^{M_j} r_{k_i\ell_i}^i=\sum_{\ell=0}^{M_j} q_{k_i\ell}^{ij},\ \forall k_i=0,\dots,\tilde{\Gamma}_i,\ \forall i\in V\ \text{s.t. } (i,j)\in E,\\
& \{r_{k\ell}^j\}_{k,\ell}\in\Delta,\ \forall j\in V,\ \{q_{k_i\ell}^{ij}\}_{k_i,\ell}\in\Delta,\ \forall(i,j)\in E,\\
& r_{k\ell}^j\le 1,\ \forall k=0,\dots,\Gamma_j,\ \ell=0,\dots,M_j,\ j\in L,\ \ r_{k\ell}^j\le 0,\ \forall k=\Gamma_j+1,\dots,\tilde{\Gamma}_j,\ \ell=0,\dots,M_j,\ j\in L
\end{aligned}
$$

with variables $\{r_{k\ell}^j\}_{k=0,\dots,\tilde{\Gamma}_j,\ell=0,\dots,M_j}^{j\in V}$ and $\{q_{k_i\ell}^{ij}\}_{k_i=0,\dots,\tilde{\Gamma}_i,\ell=0,\dots,M}^{(i,j)\in E}$.

Proof. Let $\zeta\in\{1,\dots,r\}$ and let $\{d_j^\zeta\}_{j\in L}$ be external customer demands. Note that $\tilde{\Gamma}_i\ge\Gamma_i$ for all $i\in V$ with $\tilde{\Gamma}_i>\Gamma_i$ only if $i\in L$. Thus, (EC.10) differs from (4) by considering augmented variables that are now indexed by $k=0,\dots,\tilde{\Gamma}_j$ rather than $k=0,\dots,\Gamma_j$ for $j\in L$. If we set all newly-introduced variables $r_{k\ell}^j=0$ for $k=\Gamma_j+1,\dots,\tilde{\Gamma}_j$, $\forall\ell$, $\forall j\in L$, we recover the solution to (4). This is what the constraints $r_{k\ell}^j\le 0$ for some k,ℓ,j do, as they are also nonnegative variables. The constraint $r_{k\ell}^j\le 1$ is redundant with the simplex constraint. Thus, we are simply solving (4) over an enlarged space and setting the new variables we've introduced and which we do not need to zero. □

The advantage of such a reformulation is that it enables us to make use of Proposition 5, by ensuring that, regardless of the values that $\{d_j^\zeta\}_{j\in L}$ take, the size of the linear program to solve remains the same. As $\{d_j^\zeta\}_{j\in L}$ varies, the only parameters that vary are those indicated in red in (EC.10), that is, the right hand side of the linear program. We are thus able to provide lower bounds on the optimal holding cost at $\{\hat{d}_j\}_{j\in L}$ by solving (EC.10) for appropriately set right hand side values, dictated by the values $\{d_j^1\}_{j\in L},\dots,\{d_j^r\}_{j\in L}$, extracting the associated dual variables, and leveraging Proposition 5. Note that providing such a lower bound does not require us to solve any additional linear program.

F.5. Data generation for the constraint integration in Section 6.3

Network generation. Supply chain networks G with a tree structure are generated by first using the *random_tree* function of *NetworkX* (Hagberg et al. 2008) to generate an undirected tree, then converting this to a directed tree using a breadth-first search. Trees are generated with $n\in\{20,40,60,80\}$ nodes. A node's tier is then determined through the *topological_generations* function of *NetworkX*.

GSM parameters. GSM parameters are generated as in Appendix F.2. To realize the capacity constraints, we further generate mean demands for each sink node randomly in $\{15,\dots,30\}$. Moreover, the upper bound on the capacity constraint is $B^{\max} \in \{50, 100, 150, 300, 500, 1000\}$. The maximum size of the set nodes V' to which the bound is applicable is in $\{2, 4, 6, 8, 10\}$. The nodes in V' can either be from the same tier (in which case, the actual size of V' may be smaller than the maximum size if there are not enough nodes at the randomly selected tier), or across tiers. From all eligible nodes (after a tier is chosen, if needed), the set V' is randomly selected.

Instances and solving. For each combination of parameters, we generate ten instances. Sorting out instances where the added constraint makes the problem infeasible, we retain 5,311. We then attempt to obtain the exact solution to (1)+(14) using BARON, with 48 cores and a time limit of 30 minutes. We solve the constrained problem to within a 10% optimality gap in 1,545 cases. For these instances, we also solve the original, unconstrained, problem using ($EC.4$) with Gurobi, and sort out any instance where the constrained and unconstrained solutions are equal (using the upper bound of the constrained problem when the exact solution is unavailable). Finally, we are left with 722 instances, for which we solve (4)+(15) with Gurobi to arrive at the results in Figure 10. Because we use upper bounds on the constrained problem, the optimality gap displayed in Figure 10 presents an upper bound on the true optimality gap.

E-Companion References

Blaettchen P, Calmon A, Hall G (2024) RG4SC: A random generative graph model for supply chain networks. `https://scngenerator.pythonanywhere.com/` (Accessed January 30, 2025).

Hagberg A, Swart PJ, Schult DA (2008) Exploring network structure, dynamics, and function using NetworkX. Technical report, Los Alamos National Laboratory (LANL), Los Alamos, NM (United States).

Hamann M, Strasser B (2018) Graph bisection with pareto optimization. *Journal of Experimental Algorithmics (JEA)* 23:1–34.

Humair S, Willems SP (2011) Optimizing strategic safety stock placement in general acyclic networks. *Operations Research* 59(3):781–787.

Kim J, Richard JPP, Tawarmalani M (2025) A reciprocity between tree ensemble optimization and multilinear optimization. *Operations Research* 73(5):2610–2626.

Willems SP (2008) Real-world multiechelon supply chains used for inventory optimization. *Manufacturing & Service Operations Management* 10(1):19–23.