A cardinality-indexed extended formulation with valid inequalities for unsplittable multicommodity network design

Md Shahrukh Anjum*[1], Jitamitra Desai†[2], and Ishwar Murthy‡[2]

[1]Amrut Mody School of Management, Ahmedabad University, Ahmedabad, India
[2]Decision Sciences Area, Indian Institute of Management Bangalore, Bengaluru, India

Abstract

We study the unsplittable multicommodity fixed-charge network design problem, in which every commodity follows a single path and each arc is installed at most once. Its linear relaxation is weak because a commodity may occupy a fraction of an arc. We propose an extended formulation that indexes the variables of each arc by the number of commodities it carries. Its relaxation is at least as strong as that of a compact formulation with cardinality bounds and lifted capacities, and can be strictly stronger. The convex hull of the single-arc relaxation decomposes by level and projects onto the compact single-arc closure; level counterparts of cover, configuration, k-split c-strong and packing inequalities are never weaker than the classical ones, and level-aware cut-set inequalities exploit the extended space beyond that closure. Level windows and a bound-preserving reduction keep the model small. The root bound improves on the best published bound for 27 of the 31 Canad instances and equals it on the other four. On 40 random instances the approach proves optimality for 39 against 32 for the compact formulation, with search trees 2–6 times smaller. An ablation shows that exact per-level separation and the level-aware cut-set inequalities carry the bound.

Keywords: Combinatorial optimization; Network design; Unsplittable flows; Extended formulation; Valid inequalities

1 Introduction

Let $G = (\mathcal{N}, \mathcal{A})$ be a directed graph with node set $\mathcal{N}$ and arc set $\mathcal{A}$, and let $\mathcal{K}$ be a set of commodities. Each commodity $k \in \mathcal{K}$ has an origin $o(k) \in \mathcal{N}$, a destination $t(k) \in \mathcal{N}$ and a demand $d_k \in \mathbb{Z}_{>0}$ that must be sent from $o(k)$ to $t(k)$. Installing arc $a \in \mathcal{A}$ incurs a fixed cost $f_a \geq 0$ and provides a capacity $u_a \in \mathbb{Z}_{>0}$; once installed, the arc is available to every commodity. Routing one unit of commodity k on arc a costs $c_a^k \geq 0$. Each arc is installed at most once, so the design decisions are binary (fixed-charge), and the capacity of an arc cannot be increased by installing several facilities on it, as it can in network loading. The multicommodity capacitated fixed-charge network design problem is to choose the arcs to install and to route all commodities through them at minimum total fixed and routing cost, while respecting flow balance at every node and the capacity of every arc.

*Corresponding author: shahrukh.anjum@ahduni.edu.in
†jmdesai@iimb.ac.in
‡ishwar@iimb.ac.in

In its conventional setting, the flow of a commodity may be split over several paths. In many applications, however, each commodity must follow a *single* path from its origin to its destination, which gives the *unsplittable* multicommodity capacitated network design problem (UMCND). In middle-mile and fulfillment networks, all shipments of an origin–destination pair are loaded to the same sequence of terminals (Erera et al., 2013; Greening et al., 2023). In telecommunication networks, demands are often required to use non-bifurcated routing (van Hoesel et al., 2002; Bartolini and Mingozzi, 2009).

Formulations of UMCND use design variables, which select the installed arcs, and flow variables, which route the commodities on them. We use the standard node–arc formulation. For node i and commodity k let $b_i^k = 1$ if $i = o(k)$, $b_i^k = -1$ if $i = t(k)$ and $b_i^k = 0$ otherwise, and let $\delta^+(i)$ and $\delta^-(i)$ denote the arcs leaving and entering i. We write $\mathcal{K}_a \subseteq \mathcal{K}$ for the commodities that may use arc a, and set $x_a^k = 0$ for $k \notin \mathcal{K}_a$. In general $\mathcal{K}_a = \mathcal{K}$. A smaller set arises when, for example, some commodities cannot reach a, or when commodities are restricted to path sets as in Greening et al. (2026). We assume throughout that $d_k \le u_a$ for every $k \in \mathcal{K}_a$: a commodity with $d_k > u_a$ cannot use arc a in any feasible solution, and it is removed from $\mathcal{K}_a$ in preprocessing. Every result below holds for any choice of $\mathcal{K}_a$ that satisfies this assumption. For $T \subseteq \mathcal{K}$ we write $d(T) = \sum_{k \in T} d_k$. With binary variables y_a (arc a is installed) and x_a^k (commodity k uses arc a), UMCND is stated as (Magnanti and Wong, 1984; Atamtürk and Rajan, 2002; Hewitt et al., 2013)

$$
\begin{aligned}
(\mathrm{C}) \quad \min \ & \sum_{a \in \mathcal{A}} \sum_{k \in \mathcal{K}} c_a^k d_k \, x_a^k + \sum_{a \in \mathcal{A}} f_a \, y_a && && (1)\\
\text{s.t.} \ & \sum_{a \in \delta^+(i)} x_a^k - \sum_{a \in \delta^-(i)} x_a^k = b_i^k && i \in \mathcal{N},\ k \in \mathcal{K}, && (2)\\
& \sum_{k \in \mathcal{K}_a} d_k \, x_a^k \le u_a \, y_a && a \in \mathcal{A}, && (3)\\
& x_a^k \le y_a && a \in \mathcal{A},\ k \in \mathcal{K}_a, && (4)\\
& x_a^k \in \{0,1\},\ y_a \in \{0,1\} && a \in \mathcal{A},\ k \in \mathcal{K}. && (5)
\end{aligned}
$$

The objective (1) minimizes the sum of the routing and fixed costs. For binary x, the flow balance constraints (2) decompose the flow of each commodity into an $o(k)$–$t(k)$ path and possibly directed cycles. Since routing costs are nonnegative, removing these cycles preserves flow balance and capacity feasibility and does not increase the cost. Hence some optimal solution routes every commodity on a single path. Constraints (3) enforce the arc capacity, and (5) imposes the binary restrictions. The linking inequalities (4) are implied by (3) for integer points. Their role is in the linear relaxation: they force $y_a \ge x_a^k$, whereas (3) alone allows y_a to be as small as the fraction $\sum_k d_k x_a^k / u_a$ of capacity used. Adding them is known to strengthen the compact relaxation considerably (Gendron et al., 1999).

UMCND is difficult to solve because the linear relaxation of (C) is weak. Relaxing integrality gives the same linear relaxation as for the splittable problem, while the integer optimum can only increase, so the gap that branch-and-bound must close is at least as large. The weakness has two sources. First, a fractional solution may install a fraction of an arc and pay for capacity in proportion to use, while every integer solution pays for whole arcs; the linking inequalities (4) largely repair this. Second, with unsplittable flows the capacity constraint (3) of an arc is a knapsack constraint: a commodity occupies all of its demand or none of it. The relaxation lets a large commodity occupy any fraction of an arc, and so fills residual capacity that no integer load could use. Much of the work on strengthening the relaxation therefore studies the single-arc (*arc-set*) relaxation and its valid inequalities (Brockmüller et al., 1996; Atamtürk and Rajan, 2002; van Hoesel et al., 2002;

Benhamiche et al., 2016; Chen et al., 2021; Greening et al., 2026). Even so, root gaps of up to several percent remain on standard benchmarks (Greening et al., 2026), and branch-and-bound trees for moderately sized instances are large.

This paper addresses the second source by indexing the variables of each arc by its *cardinality*, the number of commodities it carries. Once the cardinality is fixed, the loads an arc can carry form a knapsack set with a fixed number of items, and the formulation states explicitly how many commodities share the arc and how much load they can share. The compact relaxation misses this coupling: on a two-arc example, a strengthened compact relaxation leaves a gap of 25% of the optimal value, which the cardinality-indexed relaxation closes entirely (Example 1). Cardinality-indexed formulations have been used for problems with an assignment or tree structure (Murthy and Ransbotham, 2019; Di Francesco et al., 2021; Contu et al., 2026). UMCND has neither structure: commodities that pass through the same node may use different arcs, and the design is a general subnetwork. Inequalities that rely on the single-entry structure of those problems therefore do not carry over, and new arguments are needed (Section 2).

Contributions. The main contributions of this paper are as follows.

1. *Formulation.* We propose a cardinality-indexed formulation (CI) for UMCND, with a lifted capacity for each level. Its linear relaxation is at least as strong as that of a strengthened compact formulation (C^+), which already includes a cardinality bound and a lifted capacity, and it can be strictly stronger (Theorem 3 and Example 1).

2. *Single-arc structure.* We show that the convex hull of the arc-set relaxation of (CI) decomposes by level and projects onto the arc-set closure of the compact model (Theorems 4 and 6). The extended space therefore does not enlarge the single-arc closure; it changes how simply the closure can be approached. Level counterparts of cover, $(1, p)$-configuration, c-strong, k-split c-strong and SAC-Pack inequalities are never weaker than the classical inequalities and can be strictly stronger in the relaxation of (CI) (Theorem 9 and Example 2). Most are computed by sorting the demands, and we complement them with an exact per-level separation procedure (Section 4).

3. *Multi-arc inequalities.* We derive level-aware cut-set MIR inequalities that remain valid in general networks and imply the compact capacity cut-set MIR inequalities (Theorem 17). Unlike the single-arc families, they exploit the extended space beyond the compact arc-set closure (Example 4).

4. *Algorithm and computation.* Level windows and a bound-preserving reduction keep the model small (Theorems 18 and 19). On the 31 Canad instances, our root bound improves on the best published bound (Greening et al., 2026) for 27 instances and equals it on the other four. On 40 random instances, the approach proves optimality for 39 against 32 for the compact formulation, with search trees 2.2 to 5.6 times smaller. We also identify the instances on which cardinality indexing pays off (Section 7.6).

Section 2 reviews the related literature and positions this paper relative to the closest approaches. Section 3 strengthens (C), introduces the cardinality-indexed formulation and compares the linear relaxations. Sections 4 and 5 derive the single-arc and multi-arc inequalities, Section 6 describes the algorithm, Section 7 reports the computational study, and Section 8 concludes.

2 Literature review

We review the literature on capacitated network design, valid inequalities for unsplittable flows, and cardinality-indexed and other extended formulations, and we position our contribution relative to the closest approaches. Table 1 summarizes this positioning.

Capacitated network design. Multicommodity capacitated network design has a long history in transportation and telecommunications (Magnanti and Wong, 1984; Gendron et al., 1999); see Crainic et al. (2021) for a recent overview. For the splittable fixed-charge problem, exact and bounding methods combine strong relaxations with decomposition. Examples include Lagrangian relaxation within branch-and-bound (Holmberg and Yuan, 2000), Benders decomposition strengthened by metric and cut-set inequalities (Costa et al., 2009), branch-and-price-and-cut with column-and-row generation (Gendron and Larose, 2014), and cutting-plane methods based on cut-set inequalities and alternative commodity representations (Chouman et al., 2017). Cut-set inequalities, which bound the capacity installed across a cut by the demand that must cross it, are a central tool in this literature (Barahona, 1996; Bienstock and Günlük, 1996; Atamtürk, 2002; Raack et al., 2011). Much of that work concerns network loading, where the design variables count integer numbers of capacity modules. Binary design, as here, is the special case with at most one module per arc, so the capacity cut-set inequalities and their MIR strengthening remain valid with 0–1 design variables. These classical inequalities are our starting point: Section 5 develops cut-set inequalities that also exploit the cardinality information of the extended formulation.

Unsplittable flows and arc-set polyhedra. With unsplittable flows, the single-arc (arc-set) relaxation is a binary knapsack set coupled to an arc-installation variable. Brockmüller et al. (1996) introduced the c-strong inequalities for this set. Atamtürk and Rajan (2002) generalized them to k-split c-strong inequalities (also called r-split), introduced lifted knapsack cover inequalities, and showed that separating c-strong inequalities is $\mathcal{NP}$-hard. van Hoesel et al. (2002) studied the closely related edge capacity polytope, in which integer numbers of capacity modules are installed. Benhamiche et al. (2016) treated the case where several facilities may be installed on an arc but cannot be combined, and Chen et al. (2021) gave an exact separation algorithm for the arc-set polyhedron with several facility types and existing capacity. Our setting is the special case with a single facility of one type on each arc and no existing capacity, in which the design variable is binary. On the algorithmic side, Bartolini and Mingozzi (2009) developed algorithms for non-bifurcated network design, and Hewitt et al. (2013) combined path-based branch-and-price with guided search. For large logistics networks, optimization-based heuristics such as integer-programming-based local search produce high-quality solutions (Erera et al., 2013; Greening et al., 2023). Assessing the quality of such solutions on realistically sized instances remains difficult, in part because the available dual bounds are weak (Greening et al., 2026).

Closest work. Greening et al. (2026) strengthen dual bounds for unsplittable network design in which arcs receive integer multiples of a capacity unit. They reformulate the capacity variables with multiple-choice binaries, derive single-arc commodity packing (SAC-Pack) inequalities from the arc-set relaxation of this reformulation, and add metric inequalities selected by a Lagrangian procedure. Their reformulation indexes the number of installed capacity units, whereas ours indexes the number of commodities using an arc. The two indices encode different information, and they are complementary. When at most one capacity unit can be installed on an arc, as in the binary-design setting we study, their reformulation with its linking constraints reduces to the compact model

(C) (Section 3.4), whereas the commodity-cardinality index remains informative. Their published bounds on that benchmark are the main point of comparison in Section 7.2, and Theorem 13 shows that the level version of SAC-Pack implies the original.

Cardinality-indexed formulations. Indexing variables by a count of assigned items has been used for the generalized assignment problem by Murthy and Ransbotham (2019), for capacitated concentrator location by Di Francesco et al. (2021), and for the capacitated Steiner arborescence problem by Contu et al. (2026). In all three, cardinality-induced cover and $(1, k)$-configuration inequalities are used (as summarized by Contu et al., 2026). Translated to arcs of a network, the cardinality-induced covers are implied by our level packing inequalities, whose right-hand sides we compute exactly, and the configuration inequalities correspond to our level configuration inequalities, whose quantities are also computed exactly (Section 4.2). Contu et al. (2026) also propose p-arc cardinality matching inequalities across cut-sets. Their validity argument relies on the arborescence structure, in which each node is entered by exactly one arc. UMCND lacks this structure, so the argument does not carry over (Section 3.4). To our knowledge, a cardinality-indexed formulation has not been studied for multicommodity network design on general networks.

Extended formulations and disjunctive hulls. Our structural results rest on the disjunctive description of the convex hull of a union of polytopes (Balas, 1979) and on the view of extended formulations as a way to describe complicated projections (Conforti et al., 2010). Multiple-choice reformulations with the same disjunctive flavor have been used for network flow problems with piecewise-linear costs (Croxton et al., 2003) and for capacitated network design with integer capacities (Frangioni and Gendron, 2009). Our exact separation procedure follows the Fenchel cutting-plane approach of Boyd (1994), combined with sequential lifting (Wolsey, 1975).

3 Mathematical formulations

This section strengthens the compact formulation (C) of Section 1 without new variables to (C$^+$), and then introduces the cardinality-indexed formulation (CI). It compares the three linear relaxations and closes with the relation to other disaggregated formulations.

3.1 Arc cardinality and a strengthened compact formulation

Definition 1. The *cardinality* of arc a is $L_a = \max\{|T| : T \subseteq \mathcal{K}_a,\ d(T) \le u_a\}$, the largest number of commodities that can share arc a. For $p \le |\mathcal{K}_a|$ let $\sigma_a(p)$ be the sum of the p smallest demands in $\mathcal{K}_a$.

L_a is the optimal value of a knapsack problem that maximizes the number of items packed. An arc with $L_a = 0$ cannot carry any commodity and can be deleted, so we assume $L_a \ge 1$ throughout.

Lemma 1. *Sort $\mathcal{K}_a$ by non-decreasing demand. Then $L_a = \max\{p : \sigma_a(p) \le u_a\}$, so L_a is computed exactly in $O(|\mathcal{K}_a| \log |\mathcal{K}_a|)$ time.*

Proof. If $T \subseteq \mathcal{K}_a$ has $|T| = p$ and $d(T) \le u_a$, then $\sigma_a(p) \le d(T) \le u_a$, because the p smallest demands are a cheapest set of size p. Conversely, the p smallest commodities form a feasible set whenever $\sigma_a(p) \le u_a$. □

Table 1: Positioning this paper relative to selected related work. "Index" is the principal disaggregation index of the formulation, not every index of its variables. "Design" is the type of the design variable of an arc: binary (installed at most once, fixed charge) or integer (several capacity modules or facilities). "Arc-set" and "multi-arc" list the principal families of valid inequalities proposed in each paper; a dash means that the paper proposes none of that kind.

Work	Index	Problem	Design	Arc-set inequalities	Multi-arc inequalities
Atamtürk and Rajan (2002)	none (compact)	UMCND, general network	integer; binary as special case	c-strong, k-split, lifted cover	–
Chen et al. (2021)	none (compact)	UMCND, general network	integer, several facility types, existing capacity	exact separation	–
Frangioni and Gendron (2009)	capacity units	splittable MCND	integer	strong linking constraints of the 0–1 reformulation	–
Greening et al. (2026)	capacity units	UMCND, general network	integer; binary on Canad	SAC-Pack, Gen-SAC-Pack	metric
Murthy and Ransbotham (2019)	cardinality	generalized assignment	none	cardinality cover, configuration	–
Di Francesco et al. (2021)	cardinality	concentrator location	binary (facility opening)	cardinality cover, configuration	–
Contu et al. (2026)	cardinality	Steiner arborescence (tree)	binary	cardinality cover, configuration	p-arc cardinality matching
This paper	cardinality	UMCND, general network	**binary (fixed charge)**	level counterparts of valid inequalities $\pi^\top x \le \pi_0 y$, $\pi \ge 0$, in the sense of Theorem 9; exact per-level separation	level-aware cut-set MIR

Two simple strengthenings of (C) need no new variables. The first is the *cardinality inequality*

$$\sum_{k \in \mathcal{K}_a} x_a^k \le L_a \, y_a \qquad a \in \mathcal{A}. \tag{6}$$

The second replaces u_a in (3) by the *lifted capacity* $\bar{u}_a = \max\{d(T) : T \subseteq \mathcal{K}_a,\ d(T) \le u_a\} \le u_a$, the largest load the arc can actually carry. Since demands are integers, $\bar{u}_a$ is obtained by a subset-sum dynamic program in pseudo-polynomial $O(|\mathcal{K}_a|\, u_a)$ time. In Example 1 below, $u_a = 10$ and the demands are $(2, 2, 9)$, so $L_a = 2$ and $\bar{u}_a = 9 < u_a$. Both inequalities are valid for every feasible solution of (C). If $y_a = 0$, both sides vanish. If $y_a = 1$, the commodities on a form a set T with $d(T) \le u_a$, so $|T| \le L_a$ and $d(T) \le \bar{u}_a$. We denote by (C$^+$) the formulation (C) with (6) added and u_a replaced by $\bar{u}_a$ in (3). (C$^+$) has the same size as (C). We include it so that the effect of the extended formulation can be separated from improvements that are already available in the compact space, and we use it as a strengthened compact baseline in Section 7.

3.2 Cardinality-indexed extended formulation

The compact model records whether a commodity uses an arc, but not how many other commodities share it. The formulation below makes this count explicit. For each arc let $\Lambda_a = \{1, \dots, L_a\}$ be its *levels*. For $\ell \in \Lambda_a$ let binary $y_{a\ell} = 1$ if arc a is installed and carries exactly ℓ commodities, and binary $x_{a\ell}^k = 1$ if commodity k uses arc a together with $\ell - 1$ other commodities. The levels are not copies of the arc. They describe the load of a single facility, and the GUB constraint (12) below selects at most one of them. Each level has its own *level capacity*

$$u_{a\ell} = \max\{d(T) : T \subseteq \mathcal{K}_a,\ |T| = \ell,\ d(T) \le u_a\}, \tag{7}$$

which is well defined for $\ell \le L_a$ by Theorem 1 and satisfies $\sigma_a(\ell) \le u_{a\ell} \le \bar{u}_a \le u_a$. It is computed for all ℓ at once by a dynamic program over (count, load) in pseudo-polynomial $O(|\mathcal{K}_a|\, L_a\, u_a)$ time. The cardinality-indexed formulation is

$$\text{(CI)} \quad \min \sum_{a \in \mathcal{A}} \sum_{\ell \in \Lambda_a} \Big(\sum_{k \in \mathcal{K}} c_a^k d_k \, x_{a\ell}^k + f_a \, y_{a\ell} \Big) \tag{8}$$

$$\text{s.t.} \sum_{a \in \delta^+(i)} \sum_{\ell \in \Lambda_a} x_{a\ell}^k - \sum_{a \in \delta^-(i)} \sum_{\ell \in \Lambda_a} x_{a\ell}^k = b_i^k \qquad i \in \mathcal{N},\ k \in \mathcal{K}, \tag{9}$$

$$\sum_{k \in \mathcal{K}_a} d_k \, x_{a\ell}^k \le u_{a\ell} \, y_{a\ell} \qquad a \in \mathcal{A},\ \ell \in \Lambda_a, \tag{10}$$

$$\sum_{k \in \mathcal{K}_a} x_{a\ell}^k = \ell \, y_{a\ell} \qquad a \in \mathcal{A},\ \ell \in \Lambda_a, \tag{11}$$

$$\sum_{\ell \in \Lambda_a} y_{a\ell} \le 1 \qquad a \in \mathcal{A}, \tag{12}$$

$$x_{a\ell}^k \le y_{a\ell} \qquad a \in \mathcal{A},\ \ell \in \Lambda_a,\ k \in \mathcal{K}_a, \tag{13}$$

$$x_{a\ell}^k = 0 \qquad a \in \mathcal{A},\ \ell \in \Lambda_a,\ k \in \mathcal{K} \setminus \mathcal{K}_a, \tag{14}$$

$$x_{a\ell}^k \in \{0,1\},\ y_{a\ell} \in \{0,1\} \qquad a \in \mathcal{A},\ \ell \in \Lambda_a,\ k \in \mathcal{K}. \tag{15}$$

Constraints (9) are the flow conservation constraints written on the aggregated flows $\sum_\ell x_{a\ell}^k$. Constraints (10) and (11) state that an arc installed at level ℓ carries exactly ℓ commodities whose total demand fits its capacity. The GUB constraints (12) select at most one level per arc, and (13) are

the level-wise linking constraints. Constraints (14) are the counterpart of the convention $x_a^k = 0$ for $k \notin \mathcal{K}_a$ in (C): without them, a commodity outside $\mathcal{K}_a$ could use a level in (9) and (8) without being subject to (10)–(13). In an implementation these variables are simply not created, and we count only the remaining ones. (CI) has $\sum_a L_a(|\mathcal{K}_a|+1) \le |\mathcal{A}|(|\mathcal{K}|+1)\max_a L_a$ variables, that is $O(|\mathcal{A}||\mathcal{K}|^2)$ in the worst case. Section 6 shows how to keep only a small window of levels per arc without losing exactness.

Remark 1 (Exact fixing). Commodity k can occupy level ℓ of arc a only together with $\ell-1$ others, which weigh at least $\sigma_{a,-k}(\ell-1)$, the sum of the $\ell-1$ smallest demands in $\mathcal{K}_a \setminus \{k\}$. Hence $x_{a\ell}^k = 0$ may be fixed whenever $d_k + \sigma_{a,-k}(\ell-1) > u_{a\ell}$. This condition is equivalent to $d_k + \sigma_{a,-k}(\ell-1) > u_a$, that is, no load of ℓ commodities containing k fits on the arc. Indeed, $d_k + \sigma_{a,-k}(\ell-1)$ is the minimum load of an ℓ-commodity set containing k. If it is at most u_a, that set is feasible, so by (7) its load is at most $u_{a\ell}$; conversely, $u_{a\ell} \le u_a$. These fixings are valid for every feasible solution of (CI), and we apply them throughout: the linear relaxation of (CI) is understood with them in place. Fixing removes many variables at high levels, where only small commodities fit.

Proposition 2. (C) *and* (CI) *have the same optimal value. More precisely, the map*

$$\text{proj}: \ (x,y) \mapsto \Big(x_a^k = \sum_{\ell \in \Lambda_a} x_{a\ell}^k, \quad y_a = \sum_{\ell \in \Lambda_a} y_{a\ell} \Big) \tag{16}$$

has three properties:

(i) it sends every feasible solution of (CI) *to a feasible solution of* (C) *with the same cost;*

(ii) every feasible solution of (C) *in which each installed arc carries at least one commodity is the image of a feasible solution of* (CI)*;*

(iii) for every other feasible solution of (C) *there is one of this kind with no larger cost.*

Proof. (i) Let (x,y) be feasible for (CI). By (12) at most one $y_{a\ell}$ equals 1, so $y_a \in \{0,1\}$. By (13), only that level can carry flow, so $x_a^k \in \{0,1\}$ and $x_a^k \le y_a$. Flow conservation transfers unchanged, and $\sum_k d_k x_a^k = \sum_\ell \sum_k d_k x_{a\ell}^k \le \sum_\ell u_{a\ell} y_{a\ell} \le u_a y_a$. Costs coincide by construction. (ii) Conversely, let $(\tilde{x},\tilde{y})$ be feasible for (C) with $\ell_a = \sum_k \tilde{x}_a^k \ge 1$ for every installed arc. Since $d(\{k : \tilde{x}_a^k = 1\}) \le u_a$, we have $\ell_a \le L_a$ and the load is at most $u_{a\ell_a}$ by (7). Set $y_{a\ell_a} = 1$ and $x_{a\ell_a}^k = \tilde{x}_a^k$, and set all other variables to 0. Constraints (9)–(13) hold, and proj returns $(\tilde{x},\tilde{y})$. (iii) Closing an installed arc that carries no commodity keeps feasibility and does not increase the cost, because $f_a \ge 0$. If $f_a > 0$, it strictly decreases the cost, so such an arc never occurs in an optimal solution. □

Theorem 2 establishes that the integer optimal values coincide, not that the feasible sets correspond one to one: feasible solutions of (C) that install an empty arc have no preimage in (CI), and they are not needed.

3.3 Strength of the linear relaxation

Let $P_{(\mathrm{C}^+)}$ and $P_{(\mathrm{CI})}$ denote the feasible sets of the linear relaxations of (C^+) and (CI), with all binaries relaxed to $[0,1]$.

Proposition 3. $\text{proj}(P_{(\mathrm{CI})}) \subseteq P_{(\mathrm{C}^+)}$. *Consequently* $z_{\mathrm{LP}}(\mathrm{CI}) \ge z_{\mathrm{LP}}(\mathrm{C}^+) \ge z_{\mathrm{LP}}(\mathrm{C})$.

The two polyhedra live in different spaces, so the inclusion concerns the projection of $P_{(\mathrm{CI})}$ under (16), not $P_{(\mathrm{CI})}$ itself.

Proof. Let $(x,y) \in P_{(\mathrm{CI})}$ and $(\bar{x},\bar{y}) = \text{proj}(x,y)$. We check the constraints of (C^+) one by one.

(1) *Flow balance.* (9) is written on the aggregated flows $\bar{x}$, so (2) holds. By (14), $\bar{x}_a^k = 0$ for $k \notin \mathcal{K}_a$.
(2) *Linking.* Summing (13) over the levels gives $\bar{x}_a^k \le \sum_\ell y_{a\ell} = \bar{y}_a$ for $k \in \mathcal{K}_a$, which is (4).
(3) *Bounds.* From $y_{a\ell} \ge 0$ and (12), $0 \le \bar{y}_a \le 1$; with (2), $0 \le \bar{x}_a^k \le 1$.
(4) *Cardinality.* Summing (11) over the levels and using $\ell \le L_a$,

$$\sum_{k \in \mathcal{K}_a} \bar{x}_a^k = \sum_{\ell \in \Lambda_a} \ell \, y_{a\ell} \le L_a \sum_{\ell \in \Lambda_a} y_{a\ell} = L_a \bar{y}_a,$$

which is (6).
(5) *Lifted capacity.* Summing (10) over the levels and using $u_{a\ell} \le \bar{u}_a$,

$$\sum_{k \in \mathcal{K}_a} d_k \bar{x}_a^k \le \sum_{\ell \in \Lambda_a} u_{a\ell} y_{a\ell} \le \bar{u}_a \bar{y}_a.$$

Hence $(\bar{x}, \bar{y}) \in P_{(\mathrm{C}^+)}$. Since proj preserves the cost, this gives $z_{\mathrm{LP}}(\mathrm{CI}) \ge z_{\mathrm{LP}}(\mathrm{C}^+)$. The second inequality holds because (C^+) only adds valid inequalities to (C). □

The inclusion is strict in general, and the gap can be large even on a single pair of arcs.

Example 1. Let $\mathcal{N} = \{s, t\}$ with two parallel arcs a, b from s to t, $u_a = u_b = 10$, $f_a = f_b = 10$ and $c \equiv 0$. Three commodities from s to t have demands $d = (2, 2, 9)$. No arc can carry commodity 3 together with another commodity, so every feasible design opens both arcs and the optimal value is 20.

- (C): setting $x_a^k = x_b^k = \frac{1}{2}$ for every k and $y_a = y_b = 0.65$ satisfies (3) ($6.5 \le 6.5$) and (4). It is optimal: summing (3) over both arcs gives $13 \le 10(y_a + y_b)$, so the cost is at least 13. Hence $z_{\mathrm{LP}}(\mathrm{C}) = 13$.
- (C^+): here $L_a = 2$ and $\bar{u}_a = 9$. The same flows with $y_a = y_b = 0.75$ satisfy (6) ($1.5 \le 1.5$) and the lifted capacity ($6.5 \le 6.75$). It is optimal: (6) gives $\sum_k x_e^k \le 2y_e$ for each arc, and summing over both arcs, $3 \le 2(y_a + y_b)$, so the cost is at least 15. Hence $z_{\mathrm{LP}}(\mathrm{C}^+) = 15$.
- (CI): here $u_{e1} = 9$ and $u_{e2} = 4$ (the only feasible pair is $\{1, 2\}$). Level 2 cannot host commodity 3: any pair containing it weighs at least $11 > u_{e2}$, and Remark 1 fixes $x_{a2}^3 = x_{b2}^3 = 0$. The fixing is not needed for the bound, because the linear relaxation implies it: at level 2, (10) and (11) give $\sum_k d_k x_{e2}^k \le 4y_{e2}$ and $\sum_k x_{e2}^k = 2y_{e2}$, hence $\sum_k (d_k - 2) x_{e2}^k \le 0$, which forces $x_{e2}^3 = 0$. Let $S_1 = y_{a1} + y_{b1}$ and $S_2 = y_{a2} + y_{b2}$. Flow conservation at s sends each commodity across the two arcs exactly once, so summing (11) over both arcs and both levels gives $S_1 + 2S_2 = 3$. Commodity 3 must use level 1, so $x_{a1}^3 + x_{b1}^3 = 1$, and (13) gives $S_1 \ge 1$. Therefore

$$S_1 + S_2 = \frac{3 + S_1}{2} \ge 2,$$

and the cost $10(S_1 + S_2)$ is at least 20. The integer solution $y_{a1} = y_{b2} = 1$ attains this bound. Hence $z_{\mathrm{LP}}(\mathrm{CI}) = 20$, which closes the gap.

Optimality of the three LP values was also confirmed with an LP solver. Measured relative to the optimal value, the LP bound of (C^+) leaves a gap of $(20 - 15)/20 = 25\%$ (or 33% relative to the bound itself), and (CI) closes it entirely. The example shows only that the inclusion of Theorem 3 can be strict; the general comparison is Theorem 3.

The example shows the mechanism behind Theorem 3. In (C^+) the cardinality and capacity bounds act separately on aggregated quantities, so a large commodity can be spread fractionally over arcs that also host small ones. In (CI) each level couples the number of commodities with the load they can share, so a commodity that cannot travel with $\ell - 1$ others is excluded from level ℓ altogether.

3.4 Relation to other extended formulations

Disaggregating by cardinality has been used before for the generalized assignment problem (Murthy and Ransbotham, 2019), capacitated concentrator location (Di Francesco et al., 2021) and, most recently, the capacitated Steiner arborescence problem (Contu et al., 2026). In all three the structure is an assignment or a tree. Every customer is served by exactly one facility, or every node is entered by exactly one arc. UMCND has no such structure. Commodities with destinations in the same node set may enter it through different arcs, and a single path may cross the same cut several times. A commodity whose origin and destination both lie outside a node set may also pass through it. The number of commodities carried by the arcs of a cut is therefore not fixed by the origins and destinations alone. Cut-set arguments that match commodity counts across a cut, such as the p-arc cardinality matching inequalities of Contu et al. (2026), rely on this number being fixed, so their validity argument does not carry over. This structural observation explains why the argument fails; it does not by itself show that every transferred inequality is invalid. Computationally, however, a direct transfer of these inequalities cut off the optimal solution of one of the random instances of Section 7, so they are not valid for UMCND in general. The multi-arc inequalities of Section 5 are derived instead from load and count bounds that remain valid in general networks.

A second family of disaggregated formulations uses multiple-choice variables for the *capacity* installed on an arc, or for the segments of a piecewise-linear arc cost (Croxton et al., 2003; Frangioni and Gendron, 2009). Greening et al. (2026) apply this idea to UMCND with integer multiples of a capacity unit, using binaries y_{at} for "t units installed". Their index counts capacity units, while ours counts commodities. The difference matters when design is binary, as in the classical fixed-charge setting and the benchmark set of Crainic et al. (2001): there $t \in \{1\}$, so each arc has the single variable $y_{a1} = y_a$. We compare the two models constraint by constraint. Their capacity constraint becomes $\sum_k d_k x_a^k \le u_a y_a$, which is (3), and their multiple-choice constraint becomes $y_a \le 1$. Their disaggregated linking constraint for commodity k on arc a has the right-hand side $\sum_{t=t_{ak}^{\min}}^{1} y_{at}$, where $t_{ak}^{\min} = \lceil d_k/u_a \rceil$ is the number of units the commodity needs. If $d_k \le u_a$, then $t_{ak}^{\min} = 1$ and the constraint is $x_a^k \le y_a$, which is (4). If $d_k > u_a$, the sum is empty and the constraint forces $x_a^k = 0$, which is our convention of removing such a commodity from $\mathcal{K}_a$ (Section 1). Under the same admissibility assumption, the two models therefore have the same variables and constraints, written on paths in their path-based model and on arcs in the arc-based model they use for the Canad instances. Their reformulation thus adds nothing to (C) in this setting, whereas the cardinality index of (CI) remains informative. The two indices are complementary and can be combined, with variables indexed by (t, ℓ), for problems with integer capacity multiples. We leave this to future work.

4 Valid inequalities from the single-arc structure

Valid inequalities for network design are most often derived from the *arc-set* relaxation, which keeps the constraints of a single arc and drops flow conservation (Brockmüller et al., 1996; Atamtürk and Rajan, 2002; van Hoesel et al., 2002; Chen et al., 2021). In this section we study the arc-set relaxation of (CI). Its structure is finer than that of (C). Apart from the closed arc, it splits into one cardinality-constrained knapsack set per level. This gives an explicit description of its convex hull, a comparison with the arc-set closure of the compact model, an exact separation procedure, and a common framework for the inequality families we use.

4.1 Level sets and the convex hull of the arc set

Fix an arc a and drop the index a, writing $\mathcal{K}$ for $\mathcal{K}_a$, u for u_a, L for L_a and u_ℓ for $u_{a\ell}$. The arc-set relaxation of (CI) is

$$S = \Big\{(x, y) \in \{0,1\}^{\Lambda\times\mathcal{K}} \times \{0,1\}^{\Lambda} : (10)\text{–}(13) \text{ hold}\Big\}, \tag{17}$$

and for each level $\ell \in \Lambda$ we define the *level set*

$$X_\ell = \Big\{\chi \in \{0,1\}^{\mathcal{K}} : \sum_{k\in\mathcal{K}} \chi_k = \ell,\ \sum_{k\in\mathcal{K}} d_k\chi_k \le u_\ell\Big\}, \tag{18}$$

the incidence vectors of the loads that a level-ℓ copy of the arc can carry. $X_\ell \neq \emptyset$ for every $\ell \le L$ by Theorem 1, because the ℓ smallest demands fit. The point of (CI) is that S decomposes along levels. If $y = 0$ then $x = 0$, and if $y_\ell = 1$ then $x_\ell \in X_\ell$ and $x_{\ell'} = 0$ for $\ell' \neq \ell$. For a polytope P and $t \ge 0$ we write $tP = \{tp : p \in P\}$, with $0 \cdot P = \{0\}$.

Theorem 4.

$$\operatorname{conv}(S) = \Big\{(x, y) : y \ge 0,\ \sum_{\ell\in\Lambda} y_\ell \le 1,\ x_\ell \in y_\ell \operatorname{conv}(X_\ell)\ \ \forall \ell \in \Lambda\Big\}. \tag{19}$$

Equivalently, $\operatorname{conv}(S)$ *is described by* $y \ge 0$*, the GUB constraint* (12)*, and, for every level* ℓ *and every inequality* $\pi^\top\chi \le \pi_0$ *valid for* $\operatorname{conv}(X_\ell)$*, the inequality*

$$\pi^\top x_\ell \le \pi_0\, y_\ell. \tag{20}$$

Proof. Let Q denote the right-hand side of (19). S is the union of the point 0 and the sets $S_\ell = \{(x, y) : y = e_\ell,\ x_\ell \in X_\ell,\ x_{\ell'} = 0\ (\ell' \neq \ell)\}$. For $z \in \operatorname{conv}(X_\ell)$ let $P_\ell(z)$ be the point with $y = e_\ell$, $x_\ell = z$ and all other components 0. Then $\operatorname{conv}(S_\ell) = \{P_\ell(z) : z \in \operatorname{conv}(X_\ell)\}$, and every point of $\operatorname{conv}(S)$ has the form $\mu_0 \cdot 0 + \sum_\ell \mu_\ell P_\ell(z_\ell)$ with $z_\ell \in \operatorname{conv}(X_\ell)$, $\mu \ge 0$ and $\sum \mu = 1$ (Balas, 1979). ($\subseteq$) Since the sets S_ℓ occupy disjoint blocks of coordinates, such a point has $y_\ell = \mu_\ell$ and $x_\ell = \mu_\ell z_\ell$, so it lies in Q. ($\supseteq$) Let $(x, y) \in Q$. Set $\mu_\ell = y_\ell$ and $\mu_0 = 1 - \sum_\ell y_\ell \ge 0$. For $y_\ell > 0$ set $z_\ell = x_\ell/y_\ell \in \operatorname{conv}(X_\ell)$; for $y_\ell = 0$ we have $x_\ell = 0$, and $z_\ell \in \operatorname{conv}(X_\ell)$ is arbitrary. Then $(x, y) = \mu_0 \cdot 0 + \sum_\ell \mu_\ell P_\ell(z_\ell) \in \operatorname{conv}(S)$. No auxiliary variables are needed, because each level has its own variables. For the second statement, $\operatorname{conv}(X_\ell)$ is a polytope, so $\{(x_\ell, y_\ell) : y_\ell \ge 0,\ x_\ell \in y_\ell \operatorname{conv}(X_\ell)\}$ is its homogenization cone. If $\operatorname{conv}(X_\ell) = \{z : Az \le b\}$, this cone equals $\{(x_\ell, y_\ell) : Ax_\ell \le by_\ell,\ y_\ell \ge 0\}$. Indeed, for $y_\ell > 0$ this is the scaled polytope. For $y_\ell = 0$ it is the recession cone $\{x_\ell : Ax_\ell \le 0\}$ of the bounded polytope, which is $\{0\}$. □

Theorem 4 has three consequences that shape the rest of the paper.

Separation decomposes by level. To decide whether a point $(\bar{x}, \bar{y})$ with $\bar{y} \ge 0$ and $\sum_\ell \bar{y}_\ell \le 1$ lies in $\operatorname{conv}(S)$, it suffices to check two things. First, $\bar{x}_\ell = 0$ whenever $\bar{y}_\ell = 0$. Second, for each level with $\bar{y}_\ell > 0$, the scaled point $\bar{z}_\ell = \bar{x}_\ell/\bar{y}_\ell$ lies in $\operatorname{conv}(X_\ell)$. Each test is a separation problem over a single cardinality-constrained knapsack set. Section 4.4 shows that this problem can be solved exactly. Inequalities that combine several levels of the same arc, such as (23) below, are sums of inequalities (20) and add nothing to the closure. They are still useful in practice, because one combined row replaces several.

Relation to the arc-set closure of (C). Let $F = \{(x, y) \in \{0,1\}^{\mathcal{K}} \times \{0,1\} : \sum_k d_k x^k \leq uy,\ x \leq y\}$ be the arc-set relaxation of (C) studied by Atamtürk and Rajan (2002) and Chen et al. (2021). F contains one point that S does not represent: the arc installed but carrying no commodity, $(x, y) = (0, 1)$. In (CI) every installed level carries $\ell \geq 1$ commodities, so the map proj of (16) sends S onto $F^\circ = F \setminus \{(0, 1)\}$. Because proj is linear, $\text{proj}(\text{conv}(S)) = \text{conv}(F^\circ)$. Removing $(0, 1)$ costs nothing, since an installed empty arc is never better than a closed one. Its effect on the hull is a single inequality.

Lemma 5. $\text{conv}(F^\circ) = \text{conv}(F) \cap \{(x, y) : \sum_k x^k \geq y\}$.

Proof. The inclusion $\subseteq$ holds because every point of F° satisfies $\sum_k x^k \geq y$. Conversely, let $\bar{w} \in \text{conv}(F)$ satisfy $\sum_k \bar{w}^k_x \geq \bar{w}_y$. Write $\bar{w} = \mu_0(0, 0) + \mu_e(0, 1) + \sum_i \mu_i(\chi^i, 1)$, where the χ^i are distinct nonempty feasible loads and $\mu \geq 0$ sums to 1. We may assume $\mu_i > 0$ for all i. Since $\bar{w}_y = \mu_e + \sum_i \mu_i$ and $\sum_k \bar{w}^k_x = \sum_i \mu_i |\chi^i|$, the condition $\sum_k \bar{w}^k_x \geq \bar{w}_y$ reads

$$\sum_i \mu_i \big(|\chi^i| - 1\big) \ \geq \ \mu_e.$$

The left-hand side is the total weight available to absorb the weight μ_e of the installed-but-empty point $(0, 1)$: load χ^i can absorb up to $\mu_i(|\chi^i| - 1)$. We redistribute μ_e among the loads without exceeding these capacities. If $\mu_e = 0$, set $\varepsilon_i = 0$ for all i; otherwise set

$$\varepsilon_i = \mu_e \frac{\mu_i(|\chi^i| - 1)}{\sum_j \mu_j(|\chi^j| - 1)}.$$

Then $\sum_i \varepsilon_i = \mu_e$ and $0 \leq \varepsilon_i \leq \mu_i(|\chi^i| - 1)$ for every i. Set $\alpha_i = \mu_i/(\mu_i + \varepsilon_i)$. The capacity bound gives $\mu_i + \varepsilon_i \leq \mu_i |\chi^i|$, hence $1/|\chi^i| \leq \alpha_i \leq 1$. Moreover, $\sum_i (\mu_i + \varepsilon_i)(\alpha_i \chi^i, 1) = \big(\sum_i \mu_i \chi^i,\ \sum_i \mu_i + \mu_e\big)$, so

$$\bar{w} = \mu_0(0, 0) + \sum_i (\mu_i + \varepsilon_i)(\alpha_i \chi^i, 1),$$

a convex combination, since $\mu_0 + \sum_i (\mu_i + \varepsilon_i) = 1$. It remains to show that each $(\alpha_i \chi^i, 1)$ lies in $\text{conv}(F^\circ)$. Every nonempty subset of a feasible load is feasible, because demands are positive. Averaging the indicator vectors of all subsets of χ^i of size s gives $(s/|\chi^i|)\chi^i$, for $s = 1, \ldots, |\chi^i|$. Convex combinations of these points give $\alpha \chi^i$ for every $\alpha \in [1/|\chi^i|, 1]$. □

The level variables of different arcs are disjoint and are coupled only through the aggregated flows in (9), so the projection can be taken arc by arc.

Corollary 6. *Let $\mathcal{C}_{(\text{CI})}$ be the linear relaxation of* (CI) *intersected with* $\text{conv}(S_a)$ *for every arc a. Let $\mathcal{C}_{(\text{C})}$ be the linear relaxation of* (C) *intersected with* $\text{conv}(F_a)$ *and with $\sum_k x^k_a \geq y_a$ for every arc a. Then* $\text{proj}(\mathcal{C}_{(\text{CI})}) = \mathcal{C}_{(\text{C})}$.

Proof. For $\subseteq$, apply $\text{proj}(\text{conv}(S_a)) = \text{conv}(F^\circ_a)$ and Theorem 5 arc by arc. Flow conservation is written on aggregated flows in both models. For $\supseteq$, let $(\bar{x}, \bar{y}) \in \mathcal{C}_{(\text{C})}$. For each arc, $(\bar{x}_a, \bar{y}_a) \in \text{conv}(F^\circ_a) = \text{proj}(\text{conv}(S_a))$, so it has a preimage in $\text{conv}(S_a)$. These preimages use disjoint variables and together satisfy (9). □

Theorem 6 states precisely what the extended space does and does not add at the level of a single arc. The arc-set closure of (CI) is, after projection, the arc-set closure of the compact model together with the inequality $\sum_k x^k_a \geq y_a$. Both models can in principle reach this closure. In the compact model, exact separation over $\text{conv}(F)$ is possible by the same row-generation scheme, with a knapsack oracle in place of (29), or with the algorithm of Chen et al. (2021). The advantages of (CI) therefore lie elsewhere.

- Its linear relaxation already captures part of this closure at no separation cost (Theorem 3).
- Simple closed-form families in the extended space are, inequality by inequality, at least as strong as their classical counterparts, with right-hand sides that are never larger (Section 4.3). Bounds that would otherwise require exact separation are approached by inequalities that are mostly computed by sorting the demands.
- The level variables support inequalities that couple several arcs (Section 5), which lie outside any single-arc closure, and they provide additional branching dichotomies.

Classical knapsack inequalities apply per level, with tighter data. X_ℓ is contained in the knapsack set $\{\chi \in \{0,1\}^{\mathcal{K}} : d^\top \chi \le u_\ell\}$, so every inequality valid for that knapsack, such as lifted cover, c-strong or k-split c-strong, yields a valid inequality (20). They use the level capacity $u_\ell \le u$ in place of u, and by Theorem 9 they are never weaker than their compact counterparts. In addition, X_ℓ fixes the number of commodities. The families of Section 4.2 exploit this, through the commodities that must accompany a given subset at a given level.

4.2 Level-based inequality families

All families below have the form (20) for a fixed level ℓ, or are sums of such inequalities over levels. They are cheap to compute, because each requires only sorting and prefix sums. They are separated heuristically, alongside the exact procedure of Section 4.4. For $C \subseteq \mathcal{K}$ let $\sigma_C(p)$ be the sum of the p smallest demands in C, with $\sigma_C(0) = 0$ and $\sigma_C(p) = +\infty$ if $p > |C|$.

Level packing inequalities. For $C \subseteq \mathcal{K}$ and level ℓ, let

$$\beta_\ell(C) = \max\{|T \cap C| : T \in X_\ell\} = \max\left\{z \in \{0,\dots,\ell\} : \sigma_C(z) + \sigma_{\mathcal{K}\setminus C}(\ell - z) \le u_\ell\right\}, \tag{21}$$

the largest number of commodities from C that a level-ℓ load can contain. The second expression holds because the cheapest load with z members of C and $\ell - z$ non-members combines the z smallest demands of each kind. A value z is therefore attainable if and only if this cheapest load fits. Attainability is not monotone in z, because swapping a member of C for a non-member can make the load heavier. We therefore compute $\beta_\ell(C)$ by scanning z downward from $\min\{\ell, |C|\}$ and stopping at the first attainable value. Such a value exists because $X_\ell \neq \emptyset$.

Proposition 7. *For every $C \subseteq \mathcal{K}$ and level ℓ, the* level packing inequality

$$\sum_{k \in C} x^k_\ell \le \beta_\ell(C)\, y_\ell \tag{22}$$

is valid for S, and so is its combination over any set $M \subseteq \Lambda$ of levels,

$$\sum_{\ell \in M} \sum_{k \in C} x^k_\ell \le \sum_{\ell \in M} \beta_\ell(C)\, y_\ell. \tag{23}$$

Proof. $\sum_{k \in C} \chi_k \le \beta_\ell(C)$ is valid for X_ℓ by definition, so (22) is an instance of (20), and (23) is a sum of such inequalities. □

Inequality (22) contains two known families as special cases.

- With $\beta_\ell(C) = |C| - 1$ it is a cover inequality for level ℓ. The term $\sigma_{\mathcal{K}\setminus C}(\ell - |C|)$ in (21) counts the commodities outside C that must also be present when the arc carries exactly ℓ commodities. As a result, a set C that fits on the arc on its own can still be a cover at a high level. These are

the cardinality-induced covers of Contu et al. (2026), transferred to arcs of a general network. Because we compute $\beta_\ell(C)$ exactly, our right-hand side is never larger than theirs. Summing the covers of one set C over the levels of an arc gives one combined inequality. We call it a GUB cover, because the GUB constraint (12) allows at most one of its levels to be active (cf. Wolsey 1990).

- With $M = \Lambda$, (23) is the counterpart, in the extended space, of the single-arc commodity packing (SAC-Pack) inequality of Greening et al. (2026). Their coefficient α_t is the largest number of selected commodities that fit into t capacity units. Ours is computed separately for each cardinality, which is tighter: in their binary-design case only $t = 1$ exists and $\alpha_1 = \max_\ell \beta_\ell(C)$.

Level configuration inequalities. Packing inequalities give every member of C the coefficient 1. A large commodity often excludes several others, and a larger coefficient on it captures this. For $R \subseteq \mathcal{K}$, a commodity $q \notin R$ and level ℓ, let $r = \beta_\ell(R)$ and

$$\gamma = \max\{|T \cap R| : T \in X_\ell,\ q \in T\} = \max\left\{z : d_q + \sigma_R(z) + \sigma_{\mathcal{K}\setminus(R\cup\{q\})}(\ell - 1 - z) \le u_\ell\right\}, \tag{24}$$

the number of members of R that can accompany q at level ℓ. It is computed by the same downward scan. If no level-ℓ load contains q, then Remark 1 fixes $x^q_\ell = 0$, and the packing inequality for R is all that is needed. We therefore assume that some level-ℓ load contains q. Then γ is well defined and $\gamma \le r$.

Proposition 8. *If $\gamma < r$, the* level configuration inequality

$$\sum_{k\in R} x^k_\ell + (r - \gamma)\, x^q_\ell \le r\, y_\ell \tag{25}$$

is valid for S.

Proof. Let $T \in X_\ell$. If $q \notin T$, the left-hand side equals $|T\cap R| \le r$. If $q \in T$, it equals $|T\cap R|+r-\gamma \le \gamma + r - \gamma = r$. Validity for S then follows from Theorem 4. □

Suppose R fits on the arc and $R\cup\{q\}$ forms a $(1, p)$-configuration, that is, q together with any p members of R exceeds the capacity. Then $r = |R|$ and $\gamma = p-1$, and (25) becomes the inequality of Padberg (1980), with the cardinality of the level built in. Because r and γ are computed exactly, (25) is valid for every R and q. The configuration conditions matter only for facet-defining properties, which we do not pursue.

Example 2. Let $u = 60$ and let the arc be able to carry the 16 commodities with demands $(22, 15, 15, 14, 14, 13, 12, 8, 8, 7, 5, 4, 4, 4, 3, 3)$. Take $q = 1$ ($d_q = 22$), $R = \{2, \dots, 7\}$ (demands $15, 15, 14, 14, 13, 12$). In the compact model, four members of R fit on the arc ($53 \le 60$), and two of them fit alongside q ($47 \le 60$). The compact configuration inequality is therefore $\sum_{k\in R} x^k + 2x^1 \le 4y$. Now consider level $\ell = 7$. A load with z members of R also contains $7 - z$ commodities from outside R, and the cheapest such load uses the smallest of them. Four members of R no longer fit ($53 + 3 + 3 + 4 = 63 > 60$), but three do ($39 + 14 = 53$), so $\beta_7(R) = 3$ (Fig. 1). With q present, two members of R no longer fit ($22 + 25 + 14 = 61 > 60$), but one does, so $\gamma = 1$. Inequality (25) reads $\sum_{k\in R} x^k_7 + 2x^1_7 \le 3y_7$. Its right-hand side is smaller than in the compact inequality. For instance, consider the point with $\bar y_7 = 1$, $\bar x^1_7 = 0$, $\bar x^k_7 = 0.8$ for the four members of R with demands $12, 13, 14, 14$, and $\bar x^k_7 = 0.95$ for the four commodities with demands $3, 3, 4, 4$. It carries $3.2+3.8 = 7$ commodities and a load of $42.4 + 13.3 = 55.7 \le 60$. It therefore satisfies (10)–(13) at level 7 and the compact inequality ($3.2 \le 4$), but violates the level inequality ($3.2 > 3$).

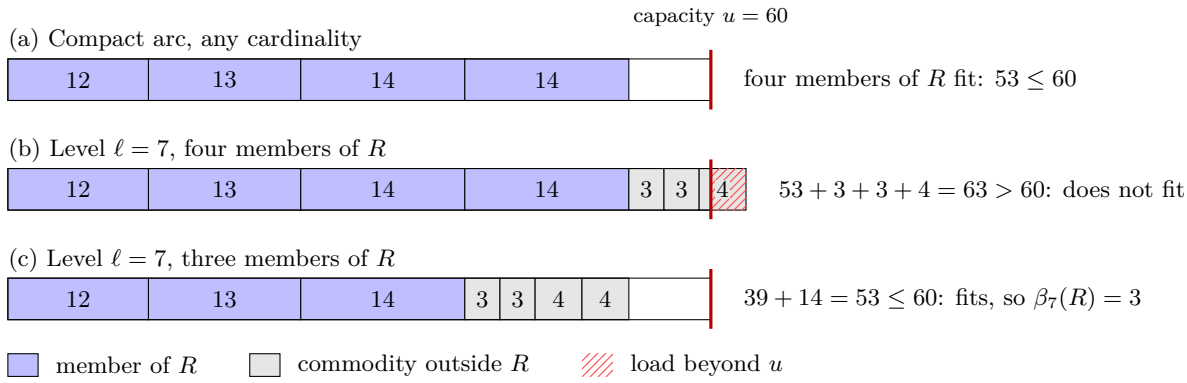


Figure 1: The load of the arc of Example 2, drawn to scale against its capacity $u = 60$. (a) Without a cardinality, the four lightest members of R fit, so the compact inequality allows four members of R. (b) At level $\ell = 7$ the arc carries exactly seven commodities; with four members of R, even the three lightest commodities outside R overload the arc. (c) With three members of R, the four lightest outsiders fit. Hence $\beta_7(R) = 3 < 4$: fixing the number of commodities lowers the right-hand side of the inequality.

4.3 Dominance over classical inequalities

The families above are cardinalized versions of known inequalities for the compact arc set F. We now show that this cardinalization never loses strength, and that each individual inequality gains strength whenever the cardinality restricts the load. The argument is general: it applies to every valid inequality of F, not only to the families we use.

Let $\pi^\top x \le \pi_0\, y$ with $\pi \ge 0$ be valid for F. Cover, $(1,p)$-configuration, lifted cover and SAC-Pack inequalities are of this form. The c-strong and k-split c-strong inequalities are written as $\pi^\top x \le c + ky$ with $c \ge 0$ and integer $k \ge 1$ (Atamtürk and Rajan, 2002). In F, $y = 0$ forces $x = 0$, so they can be homogenized to $\pi^\top x \le (c+k)\, y$. This form is valid and, since $(c+k)y \le c + ky$ for $y \le 1$, never weaker. For each level ℓ define its *level right-hand side*

$$\pi_0^\ell = \max\{\pi^\top \chi : \chi \in X_\ell\}, \tag{26}$$

and the *level counterpart* of the inequality as

$$\pi^\top x_\ell \le \pi_0^\ell\, y_\ell, \qquad \ell \in \Lambda. \tag{27}$$

Theorem 9. *Let $\pi^\top x \le \pi_0 y$ be valid for F, with $\pi \ge 0$.*

(i) Each inequality (27) is valid for S, and $\pi_0^\ell \le \pi_0$ for every ℓ.

(ii) For every (x, y) with $y \ge 0$ that satisfies all inequalities (27), the projected point $(\bar{x}, \bar{y}) = \mathrm{proj}(x, y)$ satisfies $\pi^\top \bar{x} \le \pi_0 \bar{y}$. That is, the level counterparts, together with the linear relaxation of (CI), imply the classical inequality.

(iii) If $\pi_0^\ell < \pi_0$ for some ℓ, then (27) at level ℓ strictly dominates the classical inequality restricted to that level, $\pi^\top x_\ell \le \pi_0 y_\ell$. The two have the same left-hand side, and the level inequality has the smaller right-hand side coefficient. It cuts off every point with $y_\ell > 0$ and $\pi_0^\ell y_\ell < \pi^\top x_\ell \le \pi_0 y_\ell$.

Proof. (i) $\pi^\top \chi \le \pi_0^\ell$ is valid for X_ℓ by definition, so (27) is an instance of (20). Every $\chi \in X_\ell$ satisfies $d^\top \chi \le u_\ell \le u$, so $(\chi, 1) \in F$ and $\pi^\top \chi \le \pi_0$. Hence $\pi_0^\ell \le \pi_0$. (ii) Summing (27) over ℓ and using (i), $\pi^\top \bar{x} = \sum_\ell \pi^\top x_\ell \le \sum_\ell \pi_0^\ell y_\ell \le \pi_0 \sum_\ell y_\ell = \pi_0 \bar{y}$. (iii) Immediate from (i) and $y_\ell > 0$. □

Part (iii) compares individual inequalities as halfspaces. It does not assert that the points it describes belong to the linear relaxation of (CI): the level constraints (10)–(13) may already enforce $\pi^\top x_\ell \le \pi_0^\ell y_\ell$ for a given π. Strict improvement at the level of relaxations is shown by example: Example 2 gives a point that satisfies the linear relaxation of (CI) and the classical inequality but violates a level inequality. Theorem 9 is also consistent with Theorem 6. The level counterparts of all valid inequalities for F together give, after projection, exactly the compact arc-set closure with $\sum_k x^k \ge y$, and never a strictly smaller relaxation. The comparison made by Theorem 9 is between (CI) with the level counterparts of a family $\mathcal{F}$ and (CI) (or (C^+)) with $\mathcal{F}$ itself.

Theorem 9 has an immediate consequence for bounds. Take any family $\mathcal{F}$ of arc-set inequalities used for the compact model. Adding the level counterparts of $\mathcal{F}$ to (CI) gives a bound at least as strong as adding $\mathcal{F}$ to (C^+), and hence to (C). This follows from part (ii), Theorem 3, and the fact that proj preserves cost. The gain has two sources. First, X_ℓ fixes the number of commodities, so loads with too few or too many items no longer count. Second, X_ℓ uses the level capacity $u_\ell \le u$. The families of Section 4.2 are the level counterparts of the classical families, sometimes with improved coefficients. We record the comparison family by family. Throughout, the classical quantity is the same expression with the cardinality condition removed and u_ℓ replaced by u.

Corollary 10 (Covers)**.** *Let C be a cover of the arc ($d(C) > u$), with classical cover inequality $\sum_{k\in C} x^k \le (|C|-1)\,y$. Then $\beta_\ell(C) \le \min\{\ell, |C|-1\}$ for every ℓ. The inequality is strict for $\ell < |C|-1$, and also whenever $\sigma_C(|C|-1) + \sigma_{\mathcal{K}\setminus C}(\ell - |C| + 1) > u_\ell$. In addition, a set C with $d(C) \le u$ has no classical cover inequality, yet it gives a nontrivial level packing inequality ($\beta_\ell(C) < |C|$) at every level $\ell \ge |C|$ with $\sigma_C(|C|) + \sigma_{\mathcal{K}\setminus C}(\ell - |C|) > u_\ell$.*

Corollary 11 ($(1,p)$-configurations)**.** *Let $r = \max\{|T\cap R| : d(T) \le u\}$ and $\gamma = \max\{|T\cap R| : q \in T,\ d(T) \le u\}$ be the classical quantities, with classical inequality $\sum_{k\in R} x^k + (r-\gamma)x^q \le r\,y$. Then $\beta_\ell(R) \le r$ for every ℓ. Let γ_ℓ be (24); it is defined only at levels ℓ at which some feasible load contains q, and at those levels $\gamma_\ell \le \gamma$. At the other levels, Remark 1 fixes $x_\ell^q = 0$ and the level packing inequality for R suffices. The level counterpart of the classical inequality keeps the coefficient $r-\gamma$ on x^q. At a level where q can appear, its right-hand side satisfies $\pi_0^\ell \le \max\{\beta_\ell(R),\ \gamma_\ell + r - \gamma\} \le r$: the first term bounds loads without q, and the second bounds loads with q. At a level where q cannot appear, $\pi_0^\ell \le \beta_\ell(R) \le r$. By Theorem 9(ii), these counterparts, summed over the levels, imply the classical inequality. The level configuration inequality (25) instead uses the level coefficient $\beta_\ell(R) - \gamma_\ell$ on x_ℓ^q. It is a different valid inequality, which can be stronger or weaker than the level counterpart in the coefficient of x_ℓ^q, and our separation uses both.*

Corollary 12 (c-strong and k-split c-strong)**.** *For binary design, the c-strong inequality for $S \subseteq \mathcal{K}$ is $\sum_{k\in S} x^k \le (|S| - \lceil d(S)/u\rceil + 1)\,y$ (Brockmüller et al., 1996; Atamtürk and Rajan, 2002). Its level counterpart is the level packing inequality (22) for $C = S$. That inequality satisfies $\beta_\ell(S) \le |S| - \lceil d(S)/u\rceil + 1$, because a load T with $d(T) \le u$ leaves out members of S of total demand at least $d(S) - u$. Each demand is at most u by the standing assumption of Section 1, so at least $\lceil d(S)/u\rceil - 1$ members of S are left out. The k-split inequalities have integer coefficients $\pi \ge 0$. Their level counterparts follow from (26) with the same coefficients, and they are dominated as in Theorem 9.*

Corollary 13 (SAC-Pack)**.** *With binary design, the SAC-Pack inequality of Greening et al. (2026) for $C \subseteq \mathcal{K}$ is $\sum_{k\in C} x^k \le \alpha\, y$, with $\alpha = \max\{|T\cap C| : d(T) \le u\} = \max_\ell \beta_\ell(C)$. The combined level packing inequality (23) with $M = \Lambda$ has the coefficient $\beta_\ell(C) \le \alpha$ on each y_ℓ. Since $y \ge 0$, it implies the SAC-Pack inequality under* proj*. The identity $\alpha = \max_\ell \beta_\ell(C)$ holds because every nonempty feasible load T has $|T| = \ell \le L$ for some ℓ and $d(T) \le u_\ell$.*

Table 2: Level quantities for the arc of Example 2 ($u = 60$, $R = \{2, \dots, 7\}$, $q = 1$). Classical values: $r = 4$, $\gamma = 2$. "–" means that no level-ℓ load contains q.

Level ℓ	1	2	3	4	5	6	7	8	9	10
$\beta_\ell(R)$ (classical $r = 4$)	1	2	3	4	4	4	**3**	**3**	**2**	**1**
γ_ℓ (classical $\gamma = 2$)	0	1	2	2	2	2	**1**	**1**	**0**	–

Cut-set inequalities are treated in Section 5 (Theorem 17). The remaining single-arc family, lifted cover inequalities, is covered directly by Theorem 9.

Example 3 (Example 2 continued)**.** Table 2 lists the level quantities for the arc of Example 2. Here $u = 60$, $L = 10$, $R = \{2, \dots, 7\}$ and $q = 1$. The classical cover-type inequality allows four members of R at every cardinality. With levels, four are possible only for $4 \le \ell \le 6$. At $\ell = 9$ only two fit, and at $\ell = 10$ only one. The configuration quantity γ falls in the same way, and at level 10 commodity q cannot be present at all. For instance, consider the point with $\bar{y}_9 = 1$, $\bar{x}^k_9 = 0.7$ for the three members of R with demands $12, 13, 14$, $\bar{x}^k_9 = 1$ for the six commodities with demands $3, 3, 4, 4, 4, 5$, and $\bar{x}^k_9 = 0.9$ for the commodity with demand 7. It carries $2.1 + 6 + 0.9 = 9$ commodities and a load of $56.6 \le 60$, so it satisfies (10)–(13) at level 9. It satisfies the classical inequality ($2.1 \le 4$) but violates the level-9 packing inequality ($2.1 > 2$).

4.4 Exact per-level separation

Let $(\bar{x}, \bar{y})$ satisfy $\bar{y} \ge 0$, the GUB constraint, and $\bar{x}_\ell = 0$ whenever $\bar{y}_\ell = 0$. By Theorem 4, it violates an inequality of the arc-set closure if and only if, for some level with $\bar{y}_\ell > 0$, the point $\bar{z} = \bar{x}_\ell / \bar{y}_\ell$ lies outside $\mathrm{conv}(X_\ell)$. Under the normalization $-\mathbf{1} \le \pi \le \mathbf{1}$, a most violated inequality (20) is found by solving

$$\max_{\pi, \pi_0} \ \pi^\top \bar{z} - \pi_0 \quad \text{s.t.} \quad \pi^\top \chi \le \pi_0 \ \ \forall \chi \in X_\ell, \quad -\mathbf{1} \le \pi \le \mathbf{1}. \tag{28}$$

The constraints of (28) are never enumerated. We solve it by row generation (Boyd, 1994). Starting from a few points of X_ℓ, we solve the restricted LP and obtain (π, π_0). We then call the oracle

$$\omega(\pi) = \max \Big\{ \pi^\top \chi : \chi \in \{0,1\}^{\mathcal{K}}, \ \sum_k \chi_k = \ell, \ \sum_k d_k \chi_k \le u_\ell \Big\}, \tag{29}$$

add the maximizing χ as a new row if $\omega(\pi) > \pi_0$, and repeat. With integer data, (29) is a knapsack problem with a cardinality constraint. A dynamic program over (number of items, load) solves it exactly in $O(|\mathcal{K}|\, \ell\, u_\ell)$ time and space. At termination the right-hand side is reset to the exact value $\omega(\pi)$. The inequality is therefore valid even if row generation stops early on an iteration limit.

Proposition 14. *Solved to optimality over all arcs and levels with $\bar{y}_{a\ell} > 0$, (28) either certifies that $(\bar{x}, \bar{y})$ satisfies all inequalities of the arc-set closure of* (CI)*, or returns a violated one. Row generation terminates, because X_ℓ is finite, and each oracle call takes $O(|\mathcal{K}|\, L_a\, u_a)$ time.*

Two devices keep the procedure cheap. The first trades exactness for speed.

- *Support restriction and lifting.* We optimize π only over the support $J = \{k : \bar{z}_k > 0\}$, which is typically small, and fix the other coefficients at 0. Any resulting inequality is valid, because the oracle ranges over all commodities. It may, however, miss a violated inequality that needs

nonzero coefficients outside J. Separation with this restriction is therefore a heuristic. The commodities outside J are then up-lifted sequentially (Wolsey, 1975) in a fixed order. Commodity k receives the coefficient $\pi_k = \pi_0 - \omega_k(\pi)$, where $\omega_k(\pi)$ is the value of (29) with $\chi_k = 1$ fixed, computed with the coefficients assigned so far by the same dynamic program. If no level-ℓ load contains k, then x^k_ℓ is fixed to zero by Remark 1 and no coefficient is needed. Each lifted coefficient depends on those lifted before it, so the order affects the resulting inequality, but every order yields a valid inequality that is at least as strong as the unlifted one.

- *Point pool.* X_ℓ depends only on the demands, ℓ and u_ℓ. Arcs with equal capacity share their level sets, so the points generated for one (a, ℓ) are reused as starting rows for every other pair with the same data.

In our implementation, both devices are used, together with iteration and pair limits (Section 6.2). The separation is therefore exact in principle (Theorem 14) but heuristic as run. The families of Section 4.2 are implied by the arc-set closure. We keep them because they are found by sorting alone, need no LP solve, and give sparse rows. Section 7 reports the contribution of each family.

5 Level-aware cut-set inequalities

By Theorem 6, single-arc inequalities in the extended space cannot reach beyond the arc-set closure of the compact model. Inequalities that couple several arcs can. Cut-set inequalities are the standard complement to arc-set inequalities in capacitated network design (Bienstock and Günlük, 1996; Atamtürk, 2002; Raack et al., 2011): they require the arcs leaving a node set to carry the demand that must leave it. In the compact model an open arc is credited with whatever part of that demand fits into its capacity. In (CI) the installed level also limits this credit: an arc open at level ℓ carries a load of exactly ℓ commodities, and that load can contain only part of the demand that must cross the cut. This section turns this observation into inequalities, compares them with the compact capacity cut-set MIR inequalities built on the same information (Theorem 17), and shows by example that they can close a gap that a deliberately generous compact baseline leaves open (Example 4).

For a node set $S \subset \mathcal{N}$, let $\mathcal{K}(S) = \{k : o(k) \in S,\ t(k) \notin S\}$ be the commodities that must leave S, with total demand $d(\mathcal{K}(S))$. For an arc $a \in \delta^+(S)$ let $\mathcal{K}_a(S) = \mathcal{K}(S) \cap \mathcal{K}_a$ be those of them that may use a, and let $X_{a\ell}$ denote the level set (18) of arc a. For a level $\ell \in \Lambda_a$ define

$$\kappa_{a\ell}(S) = \max\Big\{ \sum_{k \in \mathcal{K}_a(S)} d_k \chi_k : \chi \in X_{a\ell} \Big\}, \qquad \nu_{a\ell}(S) = \max\Big\{ \sum_{k \in \mathcal{K}_a(S)} \chi_k : \chi \in X_{a\ell} \Big\}, \tag{30}$$

the largest demand and the largest number of commodities of $\mathcal{K}(S)$ that a level-ℓ load of a can contain. Their compact counterparts are

$$c_a(S) = \max\{d(T) : T \subseteq \mathcal{K}_a(S),\ d(T) \le u_a\}, \qquad n_a(S) = \max\{|T| : T \subseteq \mathcal{K}_a(S),\ d(T) \le u_a\}. \tag{31}$$

A simpler bound on the level credit ignores the commodities outside $\mathcal{K}(S)$ that share the level:

$$\hat{c}_{a\ell}(S) = \max\{d(T) : T \subseteq \mathcal{K}_a(S),\ |T| \le \ell,\ d(T) \le u_{a\ell}\}. \tag{32}$$

Lemma 15. *For every S, $a \in \delta^+(S)$ and $\ell \in \Lambda_a$:*

(i) $\kappa_{a\ell}(S) \le \hat{c}_{a\ell}(S) \le c_a(S) \le \min\{\bar{u}_a, d(\mathcal{K}(S))\}$, *and* $\max_{\ell \in \Lambda_a} \kappa_{a\ell}(S) = \max_{\ell \in \Lambda_a} \hat{c}_{a\ell}(S) = c_a(S)$;

(ii) $\nu_{a\ell}(S) \le \min\{\ell, |\mathcal{K}_a(S)|\}$, $\nu_{a\ell}(S)$ *equals the quantity* $\beta_\ell(\mathcal{K}_a(S))$ *of* (21) *computed for arc* a, *and* $\max_\ell \nu_{a\ell}(S) = n_a(S)$;
(iii) if $\mathcal{K}_a(S) = \mathcal{K}_a$, *then* $\kappa_{a\ell}(S) = \hat{c}_{a\ell}(S) = u_{a\ell}$.

Proof. Let $\chi \in X_{a\ell}$ and $T = \{k \in \mathcal{K}_a(S) : \chi_k = 1\}$. Then $|T| \le \ell$ and $d(T) \le u_{a\ell}$, so $d(T) \le \hat{c}_{a\ell}(S)$ and $|T| \le \min\{\ell, |\mathcal{K}_a(S)|\}$. Every set admitted in (32) satisfies $d(T) \le u_{a\ell} \le \bar{u}_a \le u_a$, which gives $\hat{c}_{a\ell}(S) \le c_a(S) \le \bar{u}_a$; also $c_a(S) \le d(\mathcal{K}(S))$. Conversely, let $T \subseteq \mathcal{K}_a(S)$ be nonempty with $d(T) \le u_a$. Its incidence vector is a load of cardinality $|T|$, so it lies in $X_{a,|T|}$ by (7); hence $\kappa_{a,|T|}(S) \ge d(T)$ and $\nu_{a,|T|}(S) \ge |T|$, which gives the maxima. The identity with β_ℓ is (21) with $C = \mathcal{K}_a(S)$. For (iii), if $\mathcal{K}_a(S) = \mathcal{K}_a$ then $\kappa_{a\ell}(S) = \max\{d(\chi) : \chi \in X_{a\ell}\} = u_{a\ell}$ by (7), and $\hat{c}_{a\ell}(S) \le u_{a\ell}$ by definition. □

Both quantities in (30) are values of the oracle (29): $\kappa_{a\ell}(S) = \omega(\pi)$ with $\pi_k = d_k$ for $k \in \mathcal{K}_a(S)$ and $\pi_k = 0$ otherwise, and $\nu_{a\ell}(S) = \omega(\pi)$ with π the indicator vector of $\mathcal{K}_a(S)$. They are therefore computed by the same dynamic program, in $O(|\mathcal{K}_a|\,\ell\,u_{a\ell})$ time. Part (i) of Theorem 15 shows what (30) adds over (32): at level ℓ the arc carries *exactly* ℓ commodities, and those outside $\mathcal{K}(S)$, for instance commodities that pass through S, occupy capacity as well. For an arc with $u_a = 10$, $\mathcal{K}_a(S) = \{k_1\}$ with $d_{k_1} = 6$, and two further commodities of demand 5, the simpler bound gives $\hat{c}_{a2}(S) = 6$, whereas $\kappa_{a2}(S) = 0$: the only level-2 load consists of the two commodities of demand 5. By part (iii) the two coincide whenever every commodity that may use the arc must leave S.

Proposition 16. *For every* $S \subset \mathcal{N}$ *with* $\mathcal{K}(S) \neq \emptyset$, *the inequalities*

$$\sum_{a \in \delta^+(S)} \sum_{\ell \in \Lambda_a} \kappa_{a\ell}(S)\, y_{a\ell} \;\ge\; d(\mathcal{K}(S)), \tag{33}$$

$$\sum_{a \in \delta^+(S)} \sum_{\ell \in \Lambda_a} \nu_{a\ell}(S)\, y_{a\ell} \;\ge\; |\mathcal{K}(S)| \tag{34}$$

are valid for (CI). *So are the inequalities obtained by replacing* $\kappa_{a\ell}(S)$ *by* $\hat{c}_{a\ell}(S)$ *and* $\nu_{a\ell}(S)$ *by* $\min\{\ell, |\mathcal{K}_a(S)|\}$, *which are implied by them.*

Proof. In a feasible solution every $k \in \mathcal{K}(S)$ uses at least one arc of $\delta^+(S)$, because its path starts in S and ends outside S. Assign each such commodity to one arc of $\delta^+(S)$ on its path; it belongs to $\mathcal{K}_a(S)$ for that arc a. Let a be installed at level ℓ, and let $\chi \in X_{a\ell}$ be its load, by (10) and (11). The commodities assigned to a form a subset of $\{k \in \mathcal{K}_a(S) : \chi_k = 1\}$, so their demand is at most $\kappa_{a\ell}(S)$ and their number at most $\nu_{a\ell}(S)$. Summing over the arcs of $\delta^+(S)$, at most one level of each being installed by (12), gives (33) and (34). The weaker versions follow from Theorem 15(i) and (ii) and $y \ge 0$. □

Both inequalities are covering knapsack inequalities in the binary y variables, and integer rounding strengthens them. For $\theta > 0$ with $\phi = d(\mathcal{K}(S))/\theta - \lfloor d(\mathcal{K}(S))/\theta \rfloor > 0$, let

$$g_\theta(c) = \left\lfloor \frac{c}{\theta} \right\rfloor + \min\left\{1, \frac{\phi(c)}{\phi}\right\}, \qquad \phi(c) = \frac{c}{\theta} - \left\lfloor \frac{c}{\theta} \right\rfloor.$$

The mixed-integer rounding (MIR) of (33) (Marchand and Wolsey, 2001) is

$$\sum_{a \in \delta^+(S)} \sum_{\ell \in \Lambda_a} g_\theta\big(\kappa_{a\ell}(S)\big)\, y_{a\ell} \;\ge\; \left\lceil \frac{d(\mathcal{K}(S))}{\theta} \right\rceil. \tag{35}$$

Since y is binary, any coefficient larger than the right-hand side may be reduced to it. The same rounding applies to (34), with its own coefficients $\nu_{a\ell}(S)$, right-hand side $|\mathcal{K}(S)|$, and its own parameters $\theta' > 0$ and $\phi' = |\mathcal{K}(S)|/\theta' - \lfloor |\mathcal{K}(S)|/\theta' \rfloor > 0$:

$$\sum_{a\in\delta^+(S)} \sum_{\ell\in\Lambda_a} g'_{\theta'}\big(\nu_{a\ell}(S)\big)\, y_{a\ell} \;\geq\; \Big\lceil \frac{|\mathcal{K}(S)|}{\theta'} \Big\rceil, \tag{36}$$

where $g'_{\theta'}$ is g_θ with (θ, ϕ) replaced by (θ', ϕ'). A third valid inequality is the bin-packing version $\sum_{a\in\delta^+(S)} \sum_\ell y_{a\ell} \geq m(S)$. Here $m(S)$ is the largest of three lower bounds on the number of arcs of $\delta^+(S)$ that must be installed.

- $m_1(S)$ is the Martello and Toth (1990) bound L_2 for packing the demands of $\mathcal{K}(S)$ into bins of size $\max_{a\in\delta^+(S)} u_a$.
- $m_2(S)$: sort the values $c_a(S)$, $a \in \delta^+(S)$, in nonincreasing order, $c_{(1)} \geq c_{(2)} \geq \cdots$, and let $m_2(S) = \min\{m : \sum_{j=1}^m c_{(j)} \geq d(\mathcal{K}(S))\}$.
- $m_3(S)$: the same construction with the values $n_a(S)$ and target $|\mathcal{K}(S)|$.

Each installed arc carries at most $c_a(S)$ of the demand and at most $n_a(S)$ of the commodities of $\mathcal{K}(S)$, so fewer than $m_2(S)$ or $m_3(S)$ installed arcs cannot carry them all. By Theorem 15, $c_a(S) = \max_\ell \kappa_{a\ell}(S)$ and $n_a(S) = \max_\ell \nu_{a\ell}(S)$, so these are compact bounds: the bin-packing version is an inequality in the aggregated variables $y_a = \sum_\ell y_{a\ell}$ and is available in the compact model as well. We include it because it is cheap.

Comparison with compact cut-set inequalities. The fair compact counterpart of (33) credits an open arc with $c_a(S)$, the largest part of $d(\mathcal{K}(S))$ that fits into u_a. Its MIR is

$$\sum_{a\in\delta^+(S)} g_\theta\big(c_a(S)\big)\, y_a \;\geq\; \Big\lceil \frac{d(\mathcal{K}(S))}{\theta} \Big\rceil. \tag{37}$$

Because $c_a(S) \leq \bar{u}_a \leq u_a$ and g_θ is nondecreasing, (37) implies the compact capacity cut-set MIR inequalities built on $\bar{u}_a$ and on u_a with the same S and θ; the inequality with u_a is the one usually stated for binary design (Bienstock and Günlük, 1996; Atamtürk, 2002; Raack et al., 2011).

Proposition 17. *For every S with $\mathcal{K}(S) \neq \emptyset$ and every $\theta > 0$ with $\phi > 0$, inequality (35) implies, under the map* proj *of (16), the compact inequality (37), and hence the compact capacity cut-set MIR inequalities with $\bar{u}_a$ and with u_a in place of $c_a(S)$. The same holds with $\hat{c}_{a\ell}(S)$ in place of $\kappa_{a\ell}(S)$ in (35). Likewise, (36) implies the compact count MIR inequality with coefficients $g'_{\theta'}(n_a(S))$.*

Proof. The function g_θ is continuous and nondecreasing: on each interval $[j\theta, (j+1)\theta)$ it increases from j to $j+1$. By Theorem 15(i), $\kappa_{a\ell}(S) \leq \hat{c}_{a\ell}(S) \leq c_a(S)$, so with $y \geq 0$,

$$\sum_\ell g_\theta(\kappa_{a\ell}(S))\, y_{a\ell} \;\leq\; \sum_\ell g_\theta(\hat{c}_{a\ell}(S))\, y_{a\ell} \;\leq\; g_\theta(c_a(S)) \sum_\ell y_{a\ell} = g_\theta(c_a(S))\, y_a.$$

The left-hand side of (37) is therefore at least that of (35), and the right-hand sides coincide. If coefficients are reduced to the right-hand side, the same argument applies to $\min\{g_\theta(c), \lceil d(\mathcal{K}(S))/\theta \rceil\}$, which is also nondecreasing. The second chain follows from $c_a(S) \leq \bar{u}_a \leq u_a$. The count version is identical, using $\nu_{a\ell}(S) \leq n_a(S)$ from Theorem 15(ii). □

Theorem 17 holds for every instance, every S with $\mathcal{K}(S) \neq \emptyset$, and every admissible θ. Its scope is nonetheless specific. The dominance is pairwise, for the same S and θ, and holds after

projection, that is, for the implied inequality on the linear relaxation of (CI). It concerns cut-set MIR inequalities in the design variables; flow-cut-set and partial-flow variants that combine x and y (Bienstock and Günlük, 1996; Atamtürk, 2002; Raack et al., 2011) are not covered. The proposition does not assert strict improvement in general. Indeed, on a single cut crossed only by parallel arcs we found no small instance on which (35) and (36) improve on the compact arc-set closure combined with (37) and the bin-packing bound. The gain arises when an arc of the cut can be open at a level whose loads contain only part of $d(\mathcal{K}(S))$, while the compact coefficient $c_a(S)$ credits the arc with that part regardless of how many commodities it carries. Example 4 shows the mechanism.

Example 4. Let $\mathcal{N} = \{r, s, t\}$ with routing costs $c \equiv 0$ and the arcs

arc	$a_0 = (r, s)$	$a_1, a_2, a_3 = (s, t)$	$b = (r, t)$
u	40	10 each	12
f	1	10 each	10

and three commodities: k_1 and k_2 from s to t with demands 6 and 2, and k_3 from r to t with demand 3.

Integer optimum. Commodities k_1 and k_2 leave s on an arc (s, t), so at least one such arc is installed (cost 10). Commodity k_3 uses b (cost 10) or the path $r \to s \to t$. On that path it cannot share the arc of k_1 and k_2, because $6 + 2 + 3 = 11 > 10$, so it needs a second arc (s, t) (cost $1 + 10$). The optimal value is therefore 20.

Level data. The arcs (s, t) have $L = 2$ and level capacities 6 and 9. The arc b has $L_b = 3$ and $u_{b1} = 6$, $u_{b2} = 9$, $u_{b3} = 11$. Take $S = \{r, s\}$. Then $\mathcal{K}(S) = \{k_1, k_2, k_3\}$ and $d(\mathcal{K}(S)) = 11$. Every commodity is in $\mathcal{K}(S)$, so $\kappa_{a\ell}(S) = \hat{c}_{a\ell}(S) = u_{a\ell}$ by Theorem 15(iii). The compact coefficients are $c_{a_i}(S) = 9$ and $c_b(S) = 11$.

The inequalities. With $\theta = 9$ we have $\phi = 2/9$, $g_9(6) = g_9(9) = 1$ and $g_9(11) = 2$. The level-aware MIR (35) reads

$$\sum_{i=1}^{3} \left(y_{a_i 1} + y_{a_i 2}\right) + y_{b1} + y_{b2} + 2\, y_{b3} \;\geq\; 2, \tag{38}$$

whereas the compact MIR (37) with the same S and θ reads $\sum_i y_{a_i} + 2\, y_b \geq 2$. The only difference is the credit given to b. The compact inequality counts it twice whenever it is open, while (38) counts it twice only at level 3, when it carries all three commodities. At levels 1 and 2, b can carry at most 6 and 9 of the 11 units that must leave S.

LP bounds. We computed the following bounds with an LP solver:

Relaxation	Bound
LP of (C)	12
LP of (C$^+$)	16
LP of (CI)	16
compact model with the complete arc-set closure $\mathrm{conv}(F_a)$ and $\sum_k x_a^k \geq y_a$ on every arc	16
the same, plus, for *every* node set S, the convex hull of $\{y \in \{0,1\}^{\delta^+(S)} : \sum_a c_a(S) y_a \geq d(\mathcal{K}(S))\}$, the bound $m_1(S)$ and the count bound $\lceil \lvert\mathcal{K}(S)\rvert / \max_{a \in \delta^+(S)} L_a \rceil$	17.33
LP of (CI) with the single inequality (38)	20

The fifth row is a deliberately generous compact baseline: the knapsack hull implies every inequality (37), for every θ, every compact capacity cut-set MIR inequality with $\bar{u}_a$ or u_a, and the bound $m_2(S)$.

Its optimum opens a_0 with weight $2/3$, carries the loads $\{k_1, k_3\}$, $\{k_1, k_2\}$, $\{k_2, k_3\}$ on a_1, a_2, a_3 with weights $1/3, 2/3, 1/3$, and opens b with weight $1/3$, carrying k_3 alone. It satisfies the compact MIR with equality ($4/3 + 2 \cdot 1/3 = 2$). In the level space the same point has $y_{b1} = 1/3$ and all other weight of the cut at level 2, so the left-hand side of (38) is $4/3 + 1/3 = 5/3 < 2$. Adding (38) alone to the LP of (CI) closes the gap.

The example illustrates dominance over the compact capacity cut-set MIR inequalities built on $c_a(S)$, $\bar{u}_a$ or u_a, over the full single-arc closure, and over the compact bin-packing and count bounds, all for every node set. It does not claim dominance over every valid compact inequality, for instance flow-cut-set inequalities in (x, y), which we did not test.

Separation. The set S ranges over exponentially many candidates, and, as is common for cut-set inequalities (Raack et al., 2011), we search a small family of candidate sets. The candidates for S are all singletons $\{v\}$ and their complements $\mathcal{N} \setminus \{v\}$. For every origin–destination pair, we also take two minimum o–t cuts in the network with arc weights $u_a \sum_\ell \bar{y}_{a\ell}$: the one closest to o and the one closest to t. For each candidate we evaluate the MIR inequality for every θ in the set of distinct values of its coefficients at levels with $\bar{y}_{a\ell} > 0$, together with $\{u_a : a \in \delta^+(S)\}$, and add the count and bin-packing versions. This candidate list is a heuristic choice. It does not guarantee that the most violated inequality over all $\theta > 0$ is found. We keep the most violated inequality per set, and the most violated ones per round (30 in the default setting and 100 in the exhaustive setting of Section 6.2, which also adds two-node sets to the candidates). Our implementation uses the coefficients $\hat{c}_{a\ell}(S)$ of (32) and $\min\{\ell, |\mathcal{K}_a(S)|\}$, computed by the cardinality-knapsack dynamic program of (29) with a cardinality upper bound instead of an equality and, for each candidate set, cached by (u_a, ℓ); all results of Section 7 refer to this choice. The tighter coefficients (30) are values of the same oracle and can replace them at no extra cost; by Theorem 15 the two choices coincide on every arc whose admissible commodities all belong to $\mathcal{K}(S)$.

Remark 2. Unlike the cut-set matching inequalities of Contu et al. (2026), (33)–(36) do not assume that a commodity crosses a cut exactly once or that each destination has a single entering arc. Validity uses only the fact that each commodity in $\mathcal{K}(S)$ crosses $\delta^+(S)$ at least once. The inequalities are therefore valid in general networks. The remark concerns validity only. It does not claim that (33)–(36) dominate the inequalities of Contu et al. (2026) in the settings where those are valid.

6 Solution approach

The full formulation (CI) has up to L_a level variables for every arc, each with its own copy of the flow variables. On instances with many commodities this makes both the LP and the branch-and-bound tree expensive. The algorithm below works with a smaller model in three phases. Level windows restrict each arc to the levels near its LP load. They preserve the integer optimal value and give an LP bound between those of (C^+) and (CI). Cutting planes from Sections 4 and 5 are then generated at the root. Finally, a reduction returns to compact form every arc whose levels do not support the root bound, and it provably leaves the LP bound of the windowed root model, with its root inequalities, unchanged.

6.1 Level windows

For most arcs only a few levels matter, and the LP relaxation of (C) indicates roughly how many commodities an arc will carry. For each arc let $\eta_a = \min\{L_a, \lceil \sum_k \bar{x}_a^k \rceil\}$, computed from the LP optimum $\bar{x}$ of (C). The minimum is needed because (C) does not contain (6). Choose the window

$[\ell_a^-, \ell_a^+] = [\max\{1, \eta_a - \Delta\}, \min\{L_a, \eta_a + \Delta\}]$ for a width parameter Δ (we use $\Delta = 3$). The windowed formulation (CI_W) keeps one level variable for each ℓ in the window. The two end levels represent ranges of cardinalities:

- the lowest level represents $\{1, \dots, \ell_a^-\}$, and its cardinality constraint (11) becomes $y_{a\ell} \le \sum_k x_{a\ell}^k \le \ell_a^- y_{a\ell}$;
- the highest level represents $\{\ell_a^+, \dots, L_a\}$, and its constraint becomes $\ell_a^+ y_{a\ell} \le \sum_k x_{a\ell}^k \le L_a y_{a\ell}$.

Each of these is the original equation when its range is a single cardinality. If $L_a = 1$, the window is the single level 1, as in (CI). Interior levels are unchanged. The level capacity of a range is the largest $u_{a\ell}$ over the range, and the fixing rule of Remark 1 fixes $x_{a\ell}^k = 0$ only if k belongs to no load of any cardinality in the range. A range level combines several cardinalities with the largest capacity among them, so it can be more permissive than each exact level it replaces. (CI_W) is therefore a relaxation of (CI), not an exact representation of its feasible points; Theorem 18 states what it preserves.

Proposition 18. *For every choice of windows,* (CI_W) *has the same optimal value as* (C)*, and* $z_{\mathrm{LP}}(\mathrm{C}^+) \le z_{\mathrm{LP}}(\mathrm{CI}_W) \le z_{\mathrm{LP}}(\mathrm{CI})$.

Proof. Exactness. The level ranges of an arc partition $\{1, \dots, L_a\}$. The map proj of (16) sends every feasible solution of (CI_W) to a feasible solution of (C) with the same cost, by the argument of Theorem 2(i). Conversely, let a feasible solution of (C) be given in which each installed arc carries at least one commodity. Put each installed arc at the level whose range contains its number of commodities. This gives a feasible solution of (CI_W), as in the proof of Theorem 2(ii). By Theorem 2(iii) such solutions include an optimal one, so the optimal values coincide.

Upper bound. Let (x, y) be feasible for the LP of (CI). For each arc a and each range J of its window, define the range variables by summing the exact-level variables over the range, $x_{aJ}^k = \sum_{\ell \in J} x_{a\ell}^k$ and $y_{aJ} = \sum_{\ell \in J} y_{a\ell}$; an interior level is a range with one element. For the lowest range, $\sum_k x = \sum_\ell \ell y_{a\ell}$ lies between $\sum_\ell y_{a\ell}$ and $\ell_a^- \sum_\ell y_{a\ell}$, and the highest range is analogous. The range capacity is at least each $u_{a\ell}$ it replaces. A variable fixed to zero at the range is fixed at each of its levels by Remark 1, which the LP of (CI) includes, so it is zero in the summed point. All other constraints are preserved by summation, and so is the cost. The result is feasible for the LP of (CI_W), so $z_{\mathrm{LP}}(\mathrm{CI}_W) \le z_{\mathrm{LP}}(\mathrm{CI})$.

Lower bound. The argument of Theorem 3 applies unchanged. Every level of (CI_W), ranges included, satisfies $\sum_k x_{a\ell}^k \le L_a y_{a\ell}$ and has capacity at most $\bar{u}_a$. Under proj every LP point of (CI_W) therefore satisfies (6) and the lifted capacity constraint, so $z_{\mathrm{LP}}(\mathrm{C}^+) \le z_{\mathrm{LP}}(\mathrm{CI}_W)$. □

A window is too narrow when the LP optimum of (CI_W) places weight on an end level whose range contains more than one cardinality. In that case we re-center the window on the LP solution and re-solve, up to four times. The cardinality estimate η_a and the re-centering rule are heuristics. Theorem 18 holds for every window, so the choice of windows affects only the strength of the bound and the size of the model, never correctness. Re-centering is intended to improve the bound; the limit of four attempts does not guarantee that every window becomes wide enough, or that the bound of (CI_W) reaches that of (CI).

All inequalities of Sections 4 and 5 remain valid at end levels when they are computed for the level's cardinality range instead of a single cardinality. For (29) the cardinality equation becomes a range, which the dynamic program handles directly. For the cut-set inequalities of Section 5, the coefficient of a range level is the largest of the coefficients of the cardinalities it represents; for (32) and the count coefficient $\min\{\ell, |\mathcal{K}_a(S)|\}$ this means replacing ℓ by the largest cardinality of the range. Because these coefficients account for every cardinality represented by the range, the range-level inequalities remain valid; they may be weaker than their exact-level counterparts.

Inequalities whose coefficients depend on an exact count are applied only at levels that represent a single cardinality. These are (22), whose $\beta_\ell(C)$ counts the commodities outside C that must also be present, and (25).

6.2 Root cutting-plane phase

Starting from the LP optimum of (CI_W), we run rounds of separation. Each round separates the following families at the current LP point:

- level packing (22), in single-level form and combined over the levels of an arc (23), which includes the SAC-Pack counterpart of Theorem 13;
- GUB cover (Section 4.2);
- level configuration (25);
- per-level lifted cover and k-split c-strong inequalities ($k \le 4$), computed with $u_{a\ell}$ as the capacity, as in Theorems 10 and 12;
- exact per-level separation (Section 4.4);
- level-aware cut-set inequalities (33)–(35) and the bin-packing version (Section 5).

All violated inequalities found are added at once, and the LP is re-solved. The phase ends when a round finds no violated inequality, or when a maximum number of rounds is reached. For an arc a and level ℓ with $\bar{y}_{a\ell} > 0$, let $\bar{z}_k = \bar{x}^k_{a\ell}/\bar{y}_{a\ell}$ be the share of commodity k in the level, and score the pair by $\bar{y}_{a\ell} \sum_k \min\{\bar{z}_k, 1 - \bar{z}_k\}$, which measures how far the level is from an integral load. The score is a heuristic selection rule; it does not identify the most violated inequalities in any formal sense. We use two effort settings. In the *default* setting, the phase runs at most 20 rounds, exact separation is applied to the 40 pairs with the largest score, at most 60 row-generation iterations are used per pair, and at most 30 cut-set inequalities are added per round. In the *exhaustive* setting, which is used for all random instances of Section 7, the phase runs at most 50 rounds, exact separation is applied to every pair with a fractional load, at most 100 cut-set inequalities are added per round, and two-node sets are also used as cut-set candidates (Section 5). For a selected pair, the separation problem is solved exactly when the row-generation LPs and the oracle are solved to optimality within the iteration limit. The phase as a whole is heuristic, because it restricts the pairs, the iterations, the rounds and the value of k. These limits do not affect the validity of the inequalities generated, but the phase is not guaranteed to find every violated inequality of the listed families. The "root bound" reported below is the LP value of (CI_W) at the end of this phase. It contains no solver cuts.

6.3 Bound-preserving reduction before branching

After the root phase, many arcs appear in no binding inequality. Their levels enlarge every node LP but may contribute nothing to the bound. Before branch-and-bound we therefore return such arcs to compact form, and we check that the bound does not change.

For a set $\mathcal{A}^{\mathrm{lev}} \subseteq \mathcal{A}$ of arcs, the *hybrid formulation* $\mathrm{H}(\mathcal{A}^{\mathrm{lev}})$ keeps the windowed levels of every arc $a \in \mathcal{A}^{\mathrm{lev}}$. Every other arc is represented as in (C^+), by (x_a, y_a) with (3) (with $\bar{u}_a$), (4) and (6), together with $\sum_k x^k_a \ge y_a$. Flow conservation is written on the aggregated flows. By Theorem 18 applied arc by arc, $\mathrm{H}(\mathcal{A}^{\mathrm{lev}})$ has the same optimal value as (C) for every $\mathcal{A}^{\mathrm{lev}}$. Let $\mathcal{R}$ be the root inequalities, z^* the root bound, and $\mathcal{R}(\mathcal{A}^{\mathrm{lev}})$ the inequalities of $\mathcal{R}$ whose support lies in the arcs of $\mathcal{A}^{\mathrm{lev}}$. The reduction is as follows.

1. Let $\mathcal{A}^{\mathrm{lev}}$ be the set of arcs that appear in an inequality of $\mathcal{R}$ that is binding at the final root LP.
2. Solve the LP of $\mathrm{H}(\mathcal{A}^{\mathrm{lev}})$ with $\mathcal{R}(\mathcal{A}^{\mathrm{lev}})$, and let $(\hat{x}, \hat{y})$ be its optimum and $\hat{z}$ its value. If $\hat{z} \ge z^*$, stop.

3. For every compact arc $a \notin \mathcal{A}^{\text{lev}}$, solve a small LP that decides whether $(\hat{x}_a, \hat{y}_a)$ is the image under proj of a point of the arc's windowed level polytope. This polytope is defined by all constraints of (CI_W) for the arc, including the range-level cardinality bounds, the range capacities and the variable fixings, not only by the compact constraints. If the LP is feasible, it returns such a point. Add to $\mathcal{A}^{\text{lev}}$ every arc whose LP is infeasible. Lift $(\hat{x}, \hat{y})$ to the full windowed space, using the points returned for the arcs that pass. Add to $\mathcal{A}^{\text{lev}}$ the arcs of every inequality of $\mathcal{R}$ that the lifted point violates. Return to step 2.

Arcs are only ever added to $\mathcal{A}^{\text{lev}}$, never removed, so $\mathcal{A}^{\text{lev}}$ is nondecreasing throughout the procedure. The initial choice in step 1 is a heuristic; the guarantee of Theorem 19 does not depend on it, because steps 2 and 3 restore arcs until the bound is reproduced.

Theorem 19. *The reduction terminates after at most $|\mathcal{A} \setminus \mathcal{A}_0^{\text{lev}}| + 1$ solutions of the LP in step 2, where $\mathcal{A}_0^{\text{lev}}$ is the set chosen in step 1. At termination, the LP bound of $H(\mathcal{A}^{\text{lev}})$ with $\mathcal{R}(\mathcal{A}^{\text{lev}})$ equals the root bound z^*.*

Proof. First, $\hat{z} \leq z^*$ for every $\mathcal{A}^{\text{lev}}$. Let (x, y) be optimal for the LP of (CI_W) with $\mathcal{R}$. Keep its level variables on the arcs of $\mathcal{A}^{\text{lev}}$, and apply proj on the other arcs. Each such arc then satisfies (6) and the lifted capacity constraint, as in the proof of Theorem 18. It also satisfies $\sum_k x_a^k \geq y_a$, because every level of (CI_W) satisfies $\sum_k x_{a\ell}^k \geq y_{a\ell}$. The inequalities of $\mathcal{R}(\mathcal{A}^{\text{lev}})$ involve only arcs of $\mathcal{A}^{\text{lev}}$, so they still hold. The cost is unchanged, because proj preserves cost. Hence the point is feasible for the LP of $\text{H}(\mathcal{A}^{\text{lev}})$ with $\mathcal{R}(\mathcal{A}^{\text{lev}})$ and has value z^*.

Next, suppose that step 3 adds no arc to $\mathcal{A}^{\text{lev}}$. Then every compact arc passes the test, and the lifted point satisfies all of $\mathcal{R}$. It is feasible for the LP of (CI_W) with $\mathcal{R}$: on the arcs of $\mathcal{A}^{\text{lev}}$ it coincides with $(\hat{x}, \hat{y})$, on the other arcs it lies in the windowed level polytope, and its aggregated flows, which enter flow conservation, are those of $(\hat{x}, \hat{y})$. Its cost is $\hat{z}$, so $z^* \leq \hat{z}$. This contradicts the fact that the procedure did not stop in step 2. Every pass through step 3 therefore adds at least one arc to $\mathcal{A}^{\text{lev}}$. In particular, an inequality of $\mathcal{R}$ violated by the lifted point has at least one arc of its support outside $\mathcal{A}^{\text{lev}}$, since the lifted point coincides with $(\hat{x}, \hat{y})$ on $\mathcal{A}^{\text{lev}}$ and satisfies $\mathcal{R}(\mathcal{A}^{\text{lev}})$; step 3 adds every arc of its support. As $\mathcal{A}^{\text{lev}}$ is nondecreasing, at most $|\mathcal{A} \setminus \mathcal{A}_0^{\text{lev}}|$ such enlargements can occur. When $\mathcal{A}^{\text{lev}} = \mathcal{A}$, $\text{H}(\mathcal{A}^{\text{lev}})$ with $\mathcal{R}(\mathcal{A}^{\text{lev}})$ is (CI_W) with $\mathcal{R}$, and $\hat{z} = z^*$. So the procedure stops after at most $|\mathcal{A} \setminus \mathcal{A}_0^{\text{lev}}| + 1$ solutions of step 2, and at that point $\hat{z} \geq z^*$. Combined with $\hat{z} \leq z^*$, this gives $\hat{z} = z^*$. □

Theorems 18 and 19 give different guarantees. Theorem 18 concerns the integer problem: (CI_W) has the same optimal value as (C), although feasible solutions of (C) with installed but empty arcs have no preimage (they are dominated by Theorem 2(iii)). Its LP bound is not preserved in general; only $z_{\text{LP}}(\text{C}^+) \leq z_{\text{LP}}(\text{CI}_W) \leq z_{\text{LP}}(\text{CI})$ holds. Theorem 19 is an LP statement relative to (CI_W) with $\mathcal{R}$, not relative to (CI). That the reduced model has the correct integer optimum follows separately, because $\text{H}(\mathcal{A}^{\text{lev}})$ has the same optimal value as (C) and the inequalities of $\mathcal{R}$ are valid.

In the worst case the reduction returns the full windowed model. On the random instances of group D it stopped after two to four iterations and kept a mean of 93 of the 120 arcs in level form (Section 7.4).

6.4 Branch-and-bound

The reduced model $\text{H}(\mathcal{A}^{\text{lev}})$, with the inequalities of $\mathcal{R}(\mathcal{A}^{\text{lev}})$ added as constraints, is solved by Gurobi. Since these inequalities are valid for UMCND, the model has the same optimal value as (C). For each arc $a \in \mathcal{A}^{\text{lev}}$ we add a binary variable y_a defined by the equality $y_a = \sum_\ell y_{a\ell}$. By nonnegativity of the level variables, the branch $y_a = 0$ fixes every level variable of the arc to zero.

Gurobi branches first on the arc-opening variables y_a of all arcs, then on level variables, then on flows. This priority order is a design choice, motivated by the fact that opening or closing an arc is the most consequential decision and that the branch $y_a = 0$ removes all levels of the arc at once. Incumbents are harder to find on the larger model. The solve therefore starts with a phase of at most 60 s that emphasizes feasibility (MIPFocus = 1), and continues with MIPFocus = 2. This is an empirical strategy; we do not claim that it improves performance in general. All times reported in Section 7 are wall-clock times for the whole procedure: the LP of (C) used to place the windows, re-centering, the root phase, the reduction and branch-and-bound. Both methods receive the same time limit per instance group (Section 7.1).

7 Computational experiments

The computational study addresses three questions. First, how strong is the root bound of the cardinality-indexed approach compared with the best bounds published for unsplittable network design? Second, does this strength translate into more instances solved to proven optimality, and into smaller search trees, once all overhead of the disaggregated procedure is counted? Third, on which instances does cardinality indexing pay off, and on which does it not? Section 7.2 answers the first question on the standard Canad benchmark, Sections 7.3 and 7.4 answer the second on a set of random instances, and Section 7.6 draws the results together to answer the third. Section 7.5 isolates the contribution of each inequality family.

Summary of findings.

F1. Stronger root bounds.
On the 31 Canad instances the root bound is stronger than the best published bound on 27 instances and equal on the other four. It closes on average 22% of the gap that bound leaves, and 47–67% on the tight instances with 40 and 100 commodities (Section 7.2).

F2. More instances solved.
On 40 random instances the approach proves optimality for 39, against 32 for the compact formulation, and solves seven of the eight instances on which the compact formulation reaches its time limit (Sections 7.3 and 7.4).

F3. Smaller trees, faster on hard instances.
Search trees are 2.2 to 5.6 times smaller, and on the hardest group the end-to-end time, including all overhead, is about one third lower. On instances that both approaches solve in seconds, the overhead makes the compact formulation faster (Section 7.4).

F4. Two families carry the bound.
Removing the level-aware cut-set inequalities lowers the share of the compact LP gap closed from 84.6% to 72.9%, and removing exact per-level separation lowers it to 81.3%. Removing any one of the four closed-form single-arc families leaves the bound unchanged (Section 7.5).

F5. A cheap indicator.
The gains are largest when arcs carry few commodities and capacities are tight, a regime that is identified before optimization from the arc cardinalities (Section 7.6).

7.1 Experimental setup and test instances

In all experiments every commodity may use every arc, that is, $\mathcal{K}_a = \mathcal{K}$ for all $a \in \mathcal{A}$. All procedures are implemented in MATLAB R2024 and solved with Gurobi 12.0.1 on a computer with an AMD

Table 3: Average root gap (%) and gap closed on the 31 Canad class-C instances. Gaps are averages of the per-instance gaps in **??**. Unlike (39), "gap closed" in this table is measured relative to the LP bound of (C) without the linking constraints (4) (first row), and it is the average of the per-instance values. "Instances closed" counts the instances on which the bound equals z^*. None of the bounds include solver cuts.

Bound	Gap (%)	Gap closed (%)	Instances closed
LP of (C) without linking constraints (4)	18.93	–	0
LP of (C)	3.92	77.2	0
(C) + SAC-Pack (Greening et al., 2026)	2.82	85.3	2
(C) + SAC-Pack + r-split c-strong (Greening et al., 2026)	2.42	88.5	3
(CI_W)+ inequalities of Sections 4 and 5	**2.09**	**90.3**	**5**

Ryzen 7 processor (8 cores, 8 threads) and 64 GB of RAM. The Canad root bounds, which do not depend on the machine, were computed on a computing cluster. The compact formulation (C) is solved by Gurobi with default parameters. Our approach uses the settings of Section 6, with the exhaustive root-phase setting of Section 6.2. Its reported time includes model construction, window construction, cut generation, reduction and the final integer solve. Both approaches receive the same wall-clock limit per instance, which depends on the number of commodities: 300 s for $|\mathcal{K}| = 20$, 600 s for 30, 900 s for 45, 1800 s for 60 and 3600 s for 75. The limit covers the whole procedure, so the time of our approach includes all of its overhead.

We report gaps relative to the best-known objective value z^*, the best solution found by either method. An objective is called verified when at least one of the two methods proved it optimal. For a lower bound z, the gap and the share of the compact LP gap closed are

$$\mathrm{gap}(z) = 100\,\frac{z^* - z}{z^*}, \qquad \text{gap closed}(z) = 100\,\frac{z - z_{\mathrm{LP}}(\mathrm{C})}{z^* - z_{\mathrm{LP}}(\mathrm{C})}. \tag{39}$$

Gap summaries on the random instances use (39) and exclude the one instance whose best-known objective is unverified. Optimality counts use the optimality status reported by Gurobi. Runtime and node summaries below are arithmetic means; they are not shifted geometric means. Means include runs that stop at the time limit, with the time and node count at termination.

Canad instances. The 31 instances of class C of the Canad set (Crainic et al., 2001) have 20–30 nodes, 230–700 arcs and 40–400 commodities. They come with high or low fixed-to-variable cost ratios (F/V) and tight or loose capacities (T/L). They were designed for the splittable problem but remain feasible with unsplittable flows. They are the standard benchmark for the unsplittable binary-design problem (Hewitt et al., 2013; Greening et al., 2026).

7.2 Root bounds on the Canad benchmark

Table 3 compares root bounds on the Canad instances with those reported by Greening et al. (2026), whose LP values we take from their supplementary material. Their strongest configuration adds their SAC-Pack inequalities and the r-split c-strong inequalities of Atamtürk and Rajan (2002), separated to exhaustion. Like ours, their bounds contain no solver cuts. Per-instance values are in **??**.

Our root bound is stronger than the best published bound on 27 of the 31 instances and equal on the remaining four; it is never weaker. The four ties are 40-commodity instances on which both

Table 4: Root gaps on the Canad instances by number of commodities. Gaps (%) are class averages. The last two columns give the average share of the gap left by the best published bound (Greening et al., 2026) that our bound closes, for tight (T) and loose (L) capacities, over the instances on which that gap is positive (for 40 commodities, three tight instances and one loose instance).

		Root gap (%)			Remaining gap closed (%)	
$\|\mathcal{K}\|$	n	LP of (C)	Best published	Ours	T	L
40	7	4.66	0.29	0.18	66.7	15.9
100	8	4.24	2.45	1.69	46.6	25.0
200	8	4.10	4.01	3.70	12.0	4.8
400	8	2.79	2.67	2.55	6.8	4.4

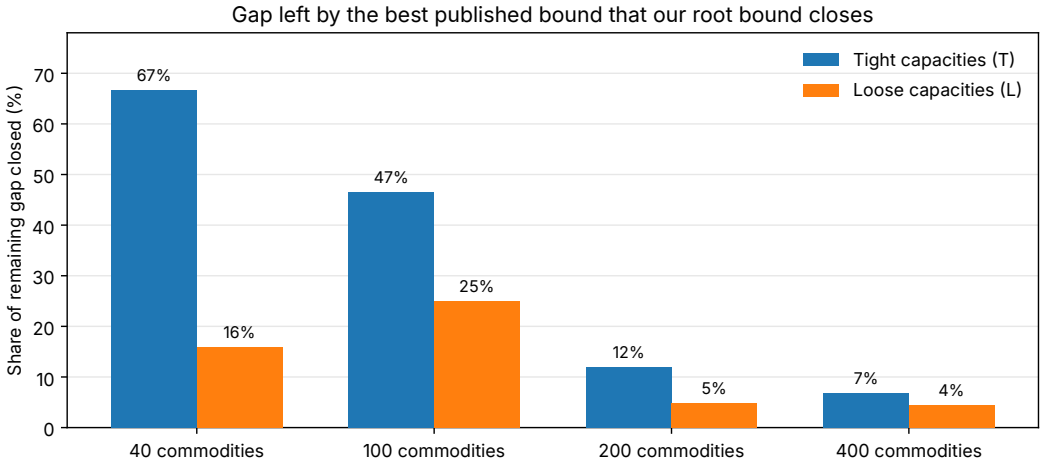


Figure 2: Share of the gap left by the best published bound (Greening et al., 2026) that our root bound closes on the Canad instances, by number of commodities and capacity tightness (class averages, Table 4).

bounds are already at or near the optimum: on three of them both equal z^*, and on 20-300-40-F-T both leave a gap of 0.21%. Our bound closes the gap at the root on five instances, against three for the best published bound. Averaged over all instances, the root gap falls from 2.42% to 2.09%. This comparison is conservative in two respects. The best published bound separates its inequalities to exhaustion, whereas our separation is heuristic as run (Section 6.2); and the bound of Greening et al. (2026) is already close to the optimum on many instances, so the remaining room for improvement is small.

A more telling measure is the share of the gap left by the best published bound that our bound closes, $100\,(z - z_{\mathrm{GDE}})/(z^* - z_{\mathrm{GDE}})$, where z_{GDE} is the bound of Greening et al. (2026). Over the 28 instances on which this gap is positive, our bound closes on average 21.9% of it. It closes it completely on 20-230-40-V-T and 20-300-40-V-T, closes 58% of it on 30-520-100-V-T and 30-700-100-V-T, and reduces the gap of 30-700-100-F-T by 1.16 percentage points, from 2.65% to 1.49%.

Where the gain comes from. The gain is far from uniform, and its pattern matches the theory. Table 4 and Fig. 2 break the results down by the number of commodities and by capacity tightness. Two regularities stand out.

First, the gain decreases with the number of commodities. With 40 and 100 commodities, the share of the remaining gap closed is 16–67%, and on the 100-commodity instances the root gap falls from 2.45% to 1.69% on average. With 200 and 400 commodities, the share falls to 4–12%. The comparison is on the same networks: 40 and 200 commodities share the 20-node networks, and 100 and 400 commodities the 30-node networks. More commodities on the same network means that each commodity is small relative to arc capacity, so an arc can carry many of them. Fixing how many commodities an arc carries then says little about the load they can share: in the terms of Theorem 9(iii), the level right-hand side π_0^ℓ is then smaller than the classical π_0 less often. With few, large commodities, the opposite holds, and the level inequalities are strictly tighter at most levels.

Second, tight capacities favor the approach in every class. For each number of commodities, the share of the remaining gap closed is larger on tight than on loose instances, by a factor of about 1.5 to 4. Tight capacities lower the cardinality L_a and make the level capacities $u_{a\ell}$ bind, which is exactly where the level structure carries information.

The 200- and 400-commodity instances also show where the approach does not help much. They retain the largest root gaps, 3.70% and 2.55% on average, and neither our inequalities nor those of Greening et al. (2026) reduce them much below the LP of (C). With many small commodities, these gaps do not stem from the single-arc knapsack structure that both families target. Inequalities that capture network-wide interactions, such as the metric inequalities of Greening et al. (2026), are complementary there.

7.3 Root bounds and solved instances on the random set

The random test set contains 40 instances in four groups of ten: A ($|\mathcal{K}| = 20$), B ($|\mathcal{K}| = 30$), C ($|\mathcal{K}| = 45$), and D ($|\mathcal{K}| = 60$–75). Group D contains the hardest instances. All groups use the same network size and the same parameter ranges, so that mainly the number of commodities changes.

Network. There are 15 nodes with coordinates drawn uniformly from $[0, 100]^2$. Five nodes are hubs, and the other ten are leaves. The hubs are pairwise connected in both directions, and every leaf is connected to and from every hub, but not to other leaves. This gives 120 directed arcs. Every leaf-to-leaf commodity then has five two-arc routes, so the network offers many alternative routes but no single bottleneck arc.

Commodities and capacities. Origins and destinations are distinct nodes drawn uniformly at random. Demands are integers drawn uniformly from $\{20, \dots, 30\}$, and capacities are integers drawn uniformly from $\{40, \dots, 80\}$. With these ranges an arc carries one to four commodities. To make instances feasible, the capacities of the arcs incident to a node are scaled up proportionally whenever their total is less than 1.5 times the demand that originates or terminates at the node. If the LP relaxation of (C) is then still infeasible, all capacities are multiplied by 1.3, at most six times.

Costs. Let h_a be the Euclidean length of arc a, $\bar{h}$ the average arc length and $\bar{u}$ the average capacity. The fixed cost is $f_a = (1 - \rho)h_a + \rho\, w\, \bar{h} \log u_a / \log \bar{u}$ with $\rho = 0.8$ and $w = 6$, so that it grows with both length and capacity. The unit routing cost is $c_a^k = \tau_a(1 + h_a/200)$ for every commodity, with τ_a drawn uniformly from $[0.05, 0.3]$. Fixed costs therefore dominate, and the design decisions drive the objective.

The average arc cardinality L_a over all arcs of the network is between 2.4 and 2.6 in every group. Since demands are at least 20 and capacities at most 80, $L_a \le 4$ on every arc. With $\Delta = 3$ the window of an arc that carries flow in the LP of (C) therefore contains every level. For an arc that is unused in that LP, the window is $\{1, 2, 3\}$ and, if $L_a = 4$, its top level represents the range $\{3, 4\}$ (Section 6.1). On these instances (CI_W) is thus close to (CI), and level windows and re-centering

Table 5: Results on the random instances. Gaps are mean percentages relative to verified best-known objectives. "Closed" is the mean of the per-instance shares of the gap of the LP of (C) closed by the final disaggregated root bound, as in (39); it therefore differs slightly from the value computed from the mean gaps. "(CI_W)" is the root gap before cuts. Times and nodes are arithmetic means and include runs stopped at the time limit. Gap statistics for group D exclude the one unverified objective.

			Root gap (%)				Solved		Time (s)		Nodes	
Group	$\lvert\mathcal{K}\rvert$	n	(C)	(CI_W)	+cuts	Closed	(C)	(CI)	(C)	(CI)	(C)	(CI)
A	20	10	12.0	5.4	2.7	76.9	10	10	5.5	11.0	494	226
B	30	10	13.6	5.8	2.8	79.7	10	10	23.0	32.6	3,240	681
C	45	10	16.2	7.2	3.0	81.1	8	10	404.9	268.0	43,932	9,333
D	60–75	10	17.2	6.4	2.6	84.7	4	9	1,225.6	790.5	81,562	14,463

matter mainly on the Canad instances.

Table 5 shows that the compact LP gap increases with instance size, from 12.0% in group A to 17.2% in group D. The final disaggregated root bound reduces the mean gap to 2.6–3.0% across groups. It closes approximately 77–85% of the compact LP gap on the instances with verified best-known objectives. For group D, this calculation uses nine verified instances; `D_harder_seed1` is excluded because its best-known objective is unverified.

Across all 40 instances, the compact formulation proves optimality for 32, while the disaggregated approach proves optimality for 39. Both methods solve all instances in groups A and B. In group C, the disaggregated approach solves all ten instances, compared with eight for the compact formulation. In group D, the disaggregated approach solves nine of ten instances, compared with four of ten for the compact formulation. The gain is therefore concentrated in the harder instances, rather than being a universal speed advantage.

Group D has a mean compact LP gap of 17.2%, a disaggregated root gap of 6.4%, and a final disaggregated root gap of 2.6%, closing approximately 84.7% of the compact LP gap across its nine verified instances. The largest share of the gap is thus closed at the difficult end of the instance set.

7.4 Branch-and-bound performance on the hard instances

The classical solver reaches its time limit on eight instances, two in group C and six in group D. The disaggregated procedure proves optimality on seven of them. In each of these seven cases its total runtime, including all preprocessing, is below the time at which the classical run stopped, so the difference in solved counts does not rest on the disaggregated procedure having been given more time. For example, `C_hard_seed1` is solved by the disaggregated procedure in 329 s, whereas the classical run stops at its limit of about 900 s, and `D_harder_seed6` is solved in 907 s against a classical limit of about 1800 s. The remaining instance, `D_harder_seed1`, reaches the time limit with both approaches; its best-known objective is unverified, and we count it as solved by neither.

Search effort. The search trees of the disaggregated approach are much smaller. The mean number of branch-and-bound nodes is smaller for the disaggregated approach in every group, by a factor of 2.2 in group A, 4.8 in B, 4.7 in C and 5.6 in D (Table 5). The reduction grows with instance difficulty. In groups C and D, the mean solution time, measured end to end and including window construction, cut generation and the reduction, is 34% and 36% lower than that of the

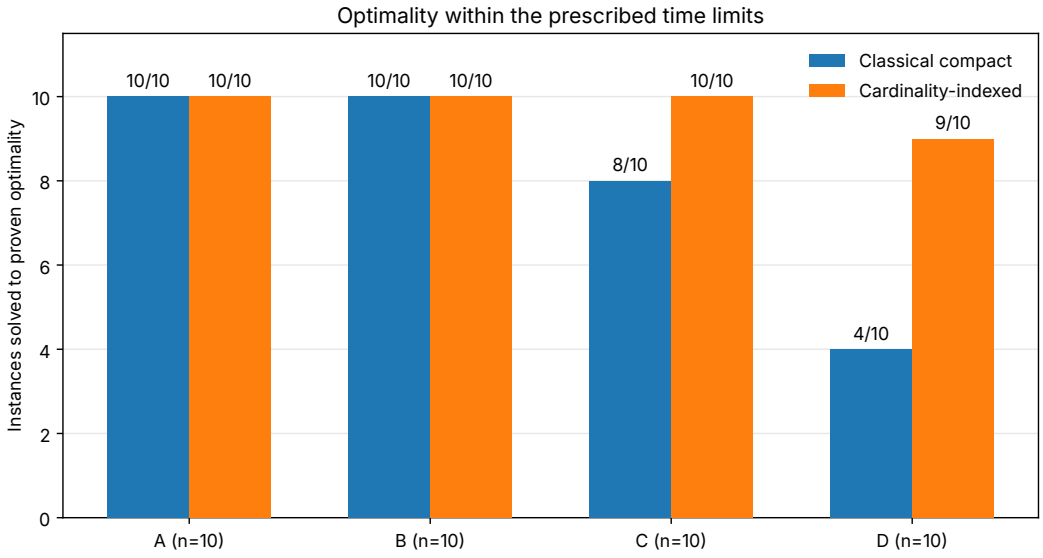


Figure 3: Number of instances solved to proven optimality within the recorded time limits. Each group contains 10 instances.

compact formulation (268.0 s against 404.9 s, and 790.5 s against 1,225.6 s). These means include runs stopped at the time limit, six of the compact formulation and one of ours in group D, so they understate the time and the nodes the compact formulation would need to prove optimality on those instances. Each node LP of the reduced model is larger than a node LP of (C), so the reduction in nodes carries over only partly into time. In groups A and B, which both approaches solve in seconds, the fixed overhead dominates and the compact formulation is faster (5.5 s against 11.0 s, and 23.0 s against 32.6 s). The smaller trees are consistent with the stronger root bounds. The ablation of Section 7.5 shows which inequalities produce those bounds, but no experiment here isolates the effect of the bound on the tree size.

Reduction statistics. On the group-D instances, the reduction retains a mean of 92.95 of the 120 arcs in disaggregated form, with a range of 77–120 arcs, and terminates after two to four iterations (mean 3.3). By Theorem 19 the root bound is unchanged by the reduction. We did not run matched experiments with the reduction disabled, so these experiments do not isolate its effect on time and nodes.

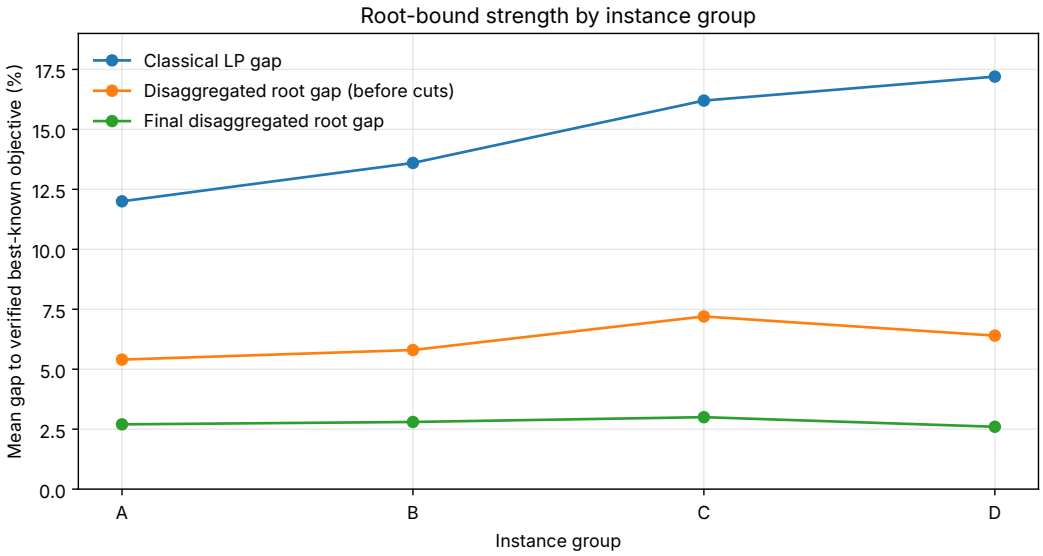


Figure 4: Mean root gaps by reported instance group: the LP of (C), the disaggregated root gap (LP of (CI_W) before cuts), and the final root gap after the cutting-plane phase. Each group has $n = 10$. Gap summaries exclude `D_harder_seed1`, whose best-known objective is unverified.

7.5 Contribution of the inequality families

The root phase combines six groups of inequalities: level packing together with GUB cover, level configuration, lifted cover, k-split c-strong, exact per-level separation and the level-aware cut-set inequalities. By Theorem 6, the four closed-form single-arc families cannot reach beyond what exact per-level separation can reach in principle, so it is an empirical question which of them matter for the bound. We therefore remove one group at a time and keep everything else fixed, including the exhaustive separation setting, the windows and the data. The test instances are the first five instances of each of the three groups with $|\mathcal{K}| = 45$, 60 and 75, that is, 15 instances. We report quantities of the root phase, which do not depend on the branching settings: the root gap relative to the best-known objective, the share of the compact LP gap closed, (39), the number of inequalities added, and the time spent on cut generation. Two instances (`D_harder_seed1` and `E_hardest_seed2`) are excluded. In some configurations the time budget of these two was used up before the root phase had run, so that their root bound is not comparable across configurations. All rows of Table 6 are averages over the same 13 instances.

The root phase accounts for most of the gain over the windowed model: it raises the share of the compact gap closed from 46.5% to 84.6% and lowers the root gap from 9.40% to 2.68%. In the full configuration it adds on average 515 inequalities per instance: 140 from exact separation, 133 k-split c-strong, 99 packing, 76 configuration, 51 lifted cover, 14 cut-set and 3 GUB cover inequalities.

The level-aware cut-set inequalities matter most. Removing them lowers the share of the gap closed by 11.7 points, to 72.9%, and raises the root gap from 2.68% to 4.74%. The final bound falls on all 13 instances, by up to 6.8% of its value. They are also the smallest family, with 14 inequalities per instance on average. This agrees with the theory: single-arc inequalities cannot go beyond the arc-set closure (Theorem 6), whereas the cut-set inequalities couple arcs and can exploit the level information beyond that closure (Example 4). The ablation removes them without

Table 6: Ablation of the inequality families on 13 random instances with $|\mathcal{K}| \in \{45, 60, 75\}$. Each row except the first three removes one group of families from the full root phase. "Root gap" is the mean gap (%) of the final root bound to the best-known objective, "Gap closed" is the mean share (%) of the compact LP gap closed, (39), "Cuts" is the mean number of inequalities added by the families of Sections 4 and 5, and "Cut-gen. time" is the mean wall-clock time of the root phase in seconds.

Configuration	Root gap (%)	Gap closed (%)	Cuts	Cut-gen. time (s)
Compact LP of (C)	17.56	0.0	–	–
Windowed model (CI_W), no inequalities	9.40	46.5	0	–
All families	2.68	84.6	515	54.4
– level packing / GUB cover	2.68	84.6	416	39.5
– level configuration	2.68	84.6	398	54.6
– lifted cover	2.68	84.6	413	48.1
– k-split c-strong	2.68	84.6	351	49.5
– exact per-level separation	3.26	81.3	1,067	95.5
– level-aware cut-set	**4.74**	**72.9**	349	31.3

substituting their compact counterparts, so it measures the value of cut-set reasoning as a whole, not of the level information within it.

Exact per-level separation comes second. Removing it lowers the share of the gap closed by 3.2 points, to 81.3%, while the number of inequalities doubles to 1,067 and the cut-generation time rises from 54 to 96 s. The closed-form families then compensate only in part, with 506 instead of 133 k-split and 374 instead of 99 packing inequalities.

The four closed-form single-arc families are redundant for the bound, one at a time. Removing level packing with GUB cover, level configuration, lifted cover or k-split c-strong leaves the final bound unchanged on every one of the 13 instances (the largest difference is below 10^{-2} in absolute value). This is what Theorem 4 leads one to expect, since exact separation covers the arc-set closure. These families still change the cost of the root phase: dropping packing and GUB cover removes 19% of the inequalities and 27% of the cut-generation time, and dropping k-split c-strong removes 32% of the inequalities. Each family was removed on its own, so the test does not exclude that they substitute for one another. We did not remove them jointly.

The effect grows with the number of commodities. By group, removing the cut-set inequalities lowers the share of the gap closed from 82.0% to 72.3% for $|\mathcal{K}| = 45$, from 85.4% to 80.5% for $|\mathcal{K}| = 60$, and from 86.9% to 66.0% for $|\mathcal{K}| = 75$. For exact separation the corresponding losses are 0.8, 0.9 and 8.5 points. Both families are thus most valuable on the largest instances, where the bound matters most. The groups contain five, four and four instances, respectively.

7.6 When does cardinality indexing pay off?

The two test sets give a consistent picture.

Where it works best. The approach is most effective when an arc can carry only a few commodities, that is, when demands are large relative to arc capacities. This is the regime of the random instances, whose average arc cardinality is 2.4–2.6. There, the final root bound closes 77–85% of the compact LP gap, the search trees are 2.2–5.6 times smaller, and seven of the eight instances on which the compact formulation reaches its time limit are solved. On the Canad instances, the same regime corresponds to 40 and 100 commodities, where our bound closes up to 67% of the gap left by the best published bound. Tight capacities strengthen the effect on both sets. The benefit

also grows with difficulty: the larger and harder the random instances, the larger the share of the gap closed (from 76.9% in group A to 84.7% in group D), and the node reduction rises from 2.2 times in group A to 4.7–5.6 times in groups B–D.

Where it helps less. When arcs carry many small commodities, as in the 200- and 400-commodity Canad instances, cardinality carries little information about load, and the bound improves by only 4–12% of the remaining gap. On easy instances that both formulations solve in seconds, the fixed cost of building the levels and generating cuts dominates, and the compact formulation is faster. Finally, the size of the extended formulation grows with the cardinalities L_a; the level windows and the reduction of Section 6 keep it manageable, at the price of a bound between those of (C^+) and (CI).

A simple indicator. The cardinality L_a of every arc is computed in $O(|\mathcal{K}| \log |\mathcal{K}|)$ time before any optimization (Theorem 1). Our results suggest that small cardinalities relative to the number of commodities, together with tight capacities, indicate instances on which the extended formulation is worth its overhead. We have not calibrated a threshold, but the indicator is available at no cost and points in the direction that both test sets confirm.

8 Conclusions and future research

We studied a cardinality-indexed formulation of unsplittable multicommodity fixed-charge network design, in which each arc is installed at most once. Indexing the arc variables by the number of commodities an arc carries splits the single-arc relaxation into one cardinality-constrained knapsack set per level, and its convex hull decomposes accordingly. This hull projects onto the arc-set closure of the compact model, together with the inequality that an installed arc carries at least one commodity. The extended space thus does not enlarge the single-arc closure. It does make the closure easier to approach. Every single-arc inequality $\pi^\top x \le \pi_0 y$ with $\pi \ge 0$, a form that includes covers, $(1, p)$-configurations, c-strong and SAC-Pack inequalities, has level counterparts that, added to the linear relaxation of the extended formulation, imply it. Each counterpart has a right-hand side no larger than the classical one, and strictly smaller whenever the cardinality of its level restricts the load. The levels also support cut-set MIR inequalities that imply the compact capacity cut-set MIR inequalities with the same cut and rounding parameter, including those built on the largest crossing demand that fits on each arc. They can close gaps that the compact model leaves open even with its complete arc-set closure and the full knapsack hull of its cut-set constraints. Level windows preserve the integer optimal value, and a reduction before branching returns arcs to compact form while provably leaving the LP bound of the windowed root model unchanged.

On the Canad benchmark, the root bound is stronger than the best published bound on 27 of 31 instances and equal on the other four, and it closes on average 22% of the gap that bound leaves. On a set of 40 random instances, the final disaggregated root bound closes approximately 77–85% of the compact LP gap across groups. More importantly, the cardinality-indexed procedure proves optimality for 39 instances, compared with 32 for the compact formulation. In group D, it solves nine of ten instances to optimality, compared with four for the compact formulation, and it proves optimality on seven of the eight instances where the classical solve reaches its time limit. The unresolved case, `D_harder_seed1`, reaches the time limit in both approaches, and its best-known objective remains unverified. The search effort drops in every group: the mean number of branch-and-bound nodes is 2.2 to 5.6 times smaller, and the reduction grows with difficulty. In the two hardest groups, the mean end-to-end solution time is about one third lower.

On the easy A and B groups, which both methods solve in seconds, the compact model is faster on average, which is consistent with the preprocessing and cut-generation overhead of the

disaggregated approach. The cardinality-indexed approach thus pays off where it matters most: on hard instances, where it proves optimality more often, with far smaller trees and in less time. The results also show where the approach is and is not worth its overhead. The benefit is greatest when only a few commodities can share an arc and capacities are tight: on the Canad instances, our bound closes 47–67% of the gap left by the best published bound on the tight instances with 40 and 100 commodities, against 4–12% on those with 200 and 400 commodities, where each arc can carry many small commodities and the cardinality carries little information about the load. Because the cardinality of every arc is computed cheaply before optimization, this regime can be recognized in advance. For practitioners, this identifies the networks in which a planner who needs certified bounds or proven-optimal designs gains most: those with few, large shipments per link and tight capacities, as in consolidation and middle-mile networks. The ablation identifies the families that carry the bound: removing the level-aware cut-set inequalities lowers the share of the compact LP gap closed from 84.6% to 72.9%, and removing exact per-level separation lowers it to 81.3%, whereas removing any one of the four closed-form single-arc families leaves the bound unchanged. Two limitations should be noted. The four closed-form families were removed one at a time, so their joint redundancy is untested, and we did not run matched experiments that isolate the effect of the reduction on time and nodes.

Three directions seem promising. First, the level index and the capacity-unit index of Greening et al. (2026) are complementary. A formulation indexed by both would extend our results to arcs that receive integer multiples of a capacity unit, as in fulfillment networks. Second, the inequalities of Sections 4 and 5 are separated only at the root. Separating them throughout the tree, in a branch-and-cut algorithm, may reduce the trees further. Third, the level variables offer branching dichotomies and primal information that the compact model lacks, and branching rules and heuristics that exploit them deserve further study.

References

Alper Atamtürk. On capacitated network design cut–set polyhedra. *Mathematical Programming*, 92:425–437, 2002.

Alper Atamtürk and Deepak Rajan. On splittable and unsplittable flow capacitated network design arc-set polyhedra. *Mathematical Programming*, 92:315–333, 2002.

Egon Balas. Disjunctive programming. *Annals of Discrete Mathematics*, 5:3–51, 1979.

Francisco Barahona. Network design using cut inequalities. *SIAM Journal on Optimization*, 6(3): 823–837, 1996.

Enrico Bartolini and Aristide Mingozzi. Algorithms for the non-bifurcated network design problem. *Journal of Heuristics*, 15(3):259–281, 2009.

Amal Benhamiche, A. Ridha Mahjoub, Nancy Perrot, and Eduardo Uchoa. Unsplittable non-additive capacitated network design using set functions polyhedra. *Computers & Operations Research*, 66:105–115, 2016.

Daniel Bienstock and Oktay Günlük. Capacitated network design—polyhedral structure and computation. *INFORMS Journal on Computing*, 8(3):243–259, 1996.

E. Andrew Boyd. Fenchel cutting planes for integer programs. *Operations Research*, 42(1):53–64, 1994.

Bodo Brockmüller, Oktay Günlük, and Laurence A. Wolsey. Designing private line networks: Polyhedral analysis and computation. CORE Discussion Paper 9647, Center for Operations Research and Econometrics, Université catholique de Louvain, 1996.

Liang Chen, Wei-Kun Chen, Mu-Ming Yang, and Yu-Hong Dai. An exact separation algorithm for unsplittable flow capacitated network design arc-set polyhedron. *Journal of Global Optimization*, 81:659–689, 2021.

Mervat Chouman, Teodor Gabriel Crainic, and Bernard Gendron. Commodity representations and cut-set-based inequalities for multicommodity capacitated fixed-charge network design. *Transportation Science*, 51(2):650–667, 2017.

Michele Conforti, Gérard Cornuéjols, and Giacomo Zambelli. Extended formulations in combinatorial optimization. *4OR*, 8(1):1–48, 2010.

Francesco Contu, Massimo Di Francesco, Enrico Gorgone, and Ishwar Murthy. An extended formulation with valid inequalities for the capacitated Steiner arborescence problem. *Networks*, 2026. doi: 10.1002/net.70053.

Alysson M. Costa, Jean-François Cordeau, and Bernard Gendron. Benders, metric and cutset inequalities for multicommodity capacitated network design. *Computational Optimization and Applications*, 42:371–392, 2009.

Teodor Gabriel Crainic, Antonio Frangioni, and Bernard Gendron. Bundle-based relaxation methods for multicommodity capacitated fixed charge network design. *Discrete Applied Mathematics*, 112(1–3):73–99, 2001.

Teodor Gabriel Crainic, Michel Gendreau, and Bernard Gendron, editors. *Network Design with Applications to Transportation and Logistics*. Springer, 2021.

Keely L. Croxton, Bernard Gendron, and Thomas L. Magnanti. Models and methods for merge-in-transit operations. *Transportation Science*, 37(1):1–22, 2003.

Massimo Di Francesco, Manlio Gaudioso, Enrico Gorgone, and Ishwar Murthy. A new extended formulation with valid inequalities for the capacitated concentrator location problem. *European Journal of Operational Research*, 289(3):975–986, 2021.

Alan Erera, Michael Hewitt, Martin Savelsbergh, and Yang Zhang. Improved load plan design through integer programming based local search. *Transportation Science*, 47(3):412–427, 2013.

Antonio Frangioni and Bernard Gendron. 0–1 reformulations of the multicommodity capacitated network design problem. *Discrete Applied Mathematics*, 157(6):1229–1241, 2009.

Bernard Gendron and Mathieu Larose. Branch-and-price-and-cut for large-scale multicommodity capacitated fixed-charge network design. *EURO Journal on Computational Optimization*, 2(1): 55–75, 2014.

Bernard Gendron, Teodor Gabriel Crainic, and Antonio Frangioni. Multicommodity capacitated network design. In B. Sansò and P. Soriano, editors, *Telecommunications Network Planning*, pages 1–19. Springer, 1999.

Lacy M. Greening, Mathieu Dahan, and Alan L. Erera. Lead-time-constrained middle-mile consolidation network design with fixed origins and destinations. *Transportation Research Part B: Methodological*, 174:102782, 2023.

Lacy M. Greening, Santanu S. Dey, and Alan L. Erera. Strengthening dual bounds for multicommodity capacitated network design with unsplittable flow constraints. *European Journal of Operational Research*, 2026. doi: 10.1016/j.ejor.2026.02.014.

Mike Hewitt, George L. Nemhauser, and Martin W. P. Savelsbergh. Branch-and-price guided search for integer programs with an application to the multicommodity fixed-charge network flow problem. *INFORMS Journal on Computing*, 25(2):302–316, 2013.

Kaj Holmberg and Di Yuan. A Lagrangian heuristic based branch-and-bound approach for the capacitated network design problem. *Operations Research*, 48(3):461–481, 2000.

Thomas L. Magnanti and Richard T. Wong. Network design and transportation planning: Models and algorithms. *Transportation Science*, 18(1):1–55, 1984.

Hugues Marchand and Laurence A. Wolsey. Aggregation and mixed integer rounding to solve MIPs. *Operations Research*, 49(3):363–371, 2001.

Silvano Martello and Paolo Toth. Lower bounds and reduction procedures for the bin packing problem. *Discrete Applied Mathematics*, 28(1):59–70, 1990.

Ishwar Murthy and Sam Ransbotham. A new extended formulation of the generalized assignment problem and some associated valid inequalities. *Discrete Applied Mathematics*, 271:119–143, 2019.

Manfred W. Padberg. $(1, k)$-configurations and facets for packing problems. *Mathematical Programming*, 18:94–99, 1980.

Christian Raack, Arie M. C. A. Koster, Sebastian Orlowski, and Roland Wessäly. On cut-based inequalities for capacitated network design polyhedra. *Networks*, 57(2):141–156, 2011.

Stan P. M. van Hoesel, Arie M. C. A. Koster, Robert L. M. J. van de Leensel, and Martin W. P. Savelsbergh. Polyhedral results for the edge capacity polytope. *Mathematical Programming*, 92: 335–358, 2002.

Laurence A. Wolsey. Faces for a linear inequality in 0–1 variables. *Mathematical Programming*, 8: 165–178, 1975.

Laurence A. Wolsey. Valid inequalities for 0–1 knapsacks and MIPs with generalised upper bound constraints. *Discrete Applied Mathematics*, 29(2–3):251–261, 1990.