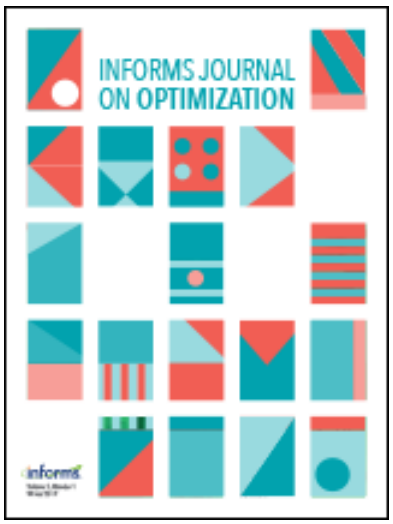


INFORMS Journal on Optimization

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Maximizing Engagement in Large-Scale Social Networks

Samuel Kroger, Hamidreza Validi, Illya V. Hicks

To cite this article:
Samuel Kroger, Hamidreza Validi, Illya V. Hicks (2024) Maximizing Engagement in Large-Scale Social Networks. INFORMS Journal on Optimization 6(3-4):196-213. https://doi.org/10.1287/ijoo.2022.0024

https://pubsonline.informs.org/journal/ijoo

INFORMS JOURNAL ON OPTIMIZATION
Vol. 6, No. 3–4, Summer–Fall 2024, pp. 196–213
ISSN 2575-1484 (print), ISSN 2575-1492 (online)

Maximizing Engagement in Large-Scale Social Networks

Samuel Kroger,[a] Hamidreza Validi,[b,*] Illya V. Hicks[a]

[a] Department of Computational Applied Mathematics and Operations Research, Rice University, Houston, Texas 77005; [b] Department of Industrial, Manufacturing and Systems Engineering, Texas Tech University, Lubbock, Texas 79409

*Corresponding author

Contact: sak8@rice.edu, https://orcid.org/0000-0001-5906-7371 (SK); hvalidi@ttu.edu, https://orcid.org/0000-0002-7983-7262 (HV); ivhicks@rice.edu, https://orcid.org/0000-0002-7815-796X (IVH)

Received: July 18, 2022
Revised: January 21, 2024; April 26, 2024
Accepted: July 4, 2024
Published Online in Articles in Advance: August 5, 2024

https://doi.org/10.1287/ijoo.2022.0024

Abstract. Motivated by the importance of user engagement as a crucial element in cascading leaving of users from a social network, we study identifying a largest relaxed variant of a degree-based cohesive subgraph: the maximum anchored k-core problem. Given graph $G = (V, E)$ and integers k and b, the maximum anchored k-core problem seeks to find a largest subset of vertices $S \subseteq V$ that induces a subgraph with at least $|S| - b$ vertices of degree at least k. We introduce a new integer programming (IP) formulation for the maximum anchored k-core problem and conduct a polyhedral study on the polytope of the problem. We show the linear programming relaxation of the proposed IP model is at least as strong as that of a naïve formulation. We also identify facet-defining inequalities of the IP formulation. Furthermore, we develop inequalities and fixing procedures to improve the computational performance of our IP model. We use benchmark instances to compare the computational performance of the IP model with (i) the naïve IP formulation and (ii) two existing heuristic algorithms. Our proposed IP model can optimally solve half of the benchmark instances that cannot be solved to optimality either by the naïve model or the existing heuristic approaches.

Funding: This work is funded by the National Science Foundation (NSF) [Grant DMS-2318790] titled AMPS: Novel Combinatorial Optimization Techniques for Smartgrids and Power Networks.
Supplemental Material: The online appendix is available at https://doi.org/10.1287/ijoo.2022.0024.

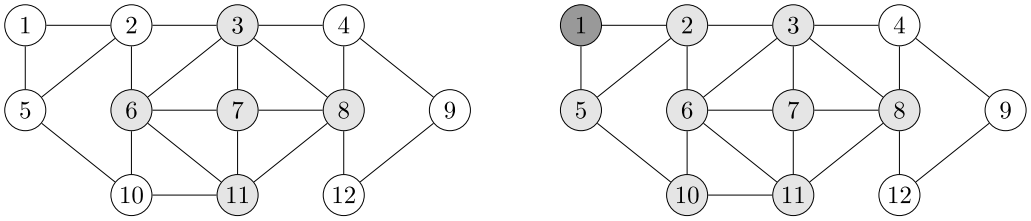
Keywords: user engagement • the maximum anchored k-core problem • integer programming

1. Introduction

Friendster was an online social networking website that launched in 2002 and attracted more than one million users in a few months (Rivlin 2006). In 2009, they began to lose active users due to multiple reasons including technical issues with their website. In 2011, Friendster discontinued its social network service after cyclical leaving patterns of its users in the Unnited States (Seki and Nakamura 2016). The failure of Friendster is closely related to poor *social resilience*—"the ability of a community to withstand external stresses and disturbances as a result of environmental changes" (Adger 2000). Garcia et al. (2013) note that the resilience of a network can be strengthened by "purchasing" some auxiliary members of the network within a limited budget. Similarly, Malliaros and Vazirgiannis (2013) and Wu et al. (2013) use *engagement* terminology to capture the interaction tendency of a user with other members of a community.[1] The problem of maximizing the engagement of a network with a limited budget can be mathematically modeled by the maximum anchored k-core problem that was introduced by Bhawalkar et al. (2015). The maximum anchored k-core problem identifies the vertices that are most crucial to forming the largest cohesive groups with respect to k-core. Given graph $G = (V, E)$ and integers k and b, an anchored k-core is a subset of vertices $S \subseteq V$ that induces a subgraph with at least $|S| - b$ vertices of degree at least k. We note that k-core and anchored k-core are combinatorially equivalent when $b = 0$. Although the maximum k-core problem is easy to be solved for any k, the maximum anchored k-core problem is NP-hard when $k \geq 3$ (Bhawalkar et al. 2015).

The operations research community might be interested in the following application of the maximum anchored k-core problem. The INFORMS Annual Meeting 2022[2] hosted a new type of 75-minute "flash" sessions in which 9–10 people present their research work. To encourage people to attend this new type of sessions, session/cluster chairs could invite a cohesive group of researchers who know at least a specific number of people, say three, in each session. Let Figure 1 be a social network of researchers who are working in a specific research area, say network optimization. Furthermore, we assume that each researcher agrees to give a talk in a flash session if they know at least three colleagues in the session. Then, gray vertices on the left side of Figure 1 represent the (maximum) three-core of the network that may be interested in presenting their works in a flash session. If the session/cluster chair convinces researcher 1

Figure 1. Social Network of Researchers



Note. (Left) The maximum three-core; (right) the maximum anchored three-core with budget $b = 1$.

to present their work at the session, then researchers 2, 5, and 10 will also be convinced to present their research in the same session. The colored vertices on the right side of Figure 1 represent the maximum anchored three-core of the social network of researchers with budget $b = 1$.

Our Contributions

In this paper, we introduce an integer programming (IP) formulation along with valid and supervalid inequalities and fixing procedures for solving the maximum anchored k-core problem. In Section 2, we provide a literature review on the maximum k-core problem and its hard variants. Section 3 introduces notation and definitions that are used throughout the paper. Section 4 proposes an IP formulation for the problem and shows that the linear programming (LP) relaxation of the model is at least as strong as that of a naïve one. In Section 5, we conduct a polyhedral study on the polytope of the problem. Section 6 introduces valid and supervalid inequalities and fixing procedures to improve the computational performance of the IP formulation. Section 7 provides an extensive set of experiments on two sets of benchmark instances. We conclude the paper in Section 8.

2. Literature Review

Identifying cohesive clusters is an important task in network analysis with a wide range of applications in marketing (Algaradi et al. 2017), social media (Pei et al. 2014), clustering and community detection (Giatsidis et al. 2011a), biology (Bader and Hogue 2002, Altaf-Ul-Amine et al. 2003), and economics (Burleson-Lesser et al. 2020). Cohesive clusters can be classified based on (i) the distance between the vertices inside clusters (Verma et al. 2015, Pajouh et al. 2016, Salemi and Buchanan 2020, Daemi et al. 2022) (e.g., cliques, k-clubs, and k-cliques); (ii) the degree of vertices in a cluster (Balasundaram et al. 2011, Ma et al. 2016, Ma and Balasundaram 2019) (e.g., k-core and k-plex); (iii) the number of edges in a cluster (Gao et al. 2022) (e.g., k-defective clique); and (iv) density (Miao and Balasundaram 2020) (e.g., quasi-clique). Interested readers are encouraged to refer to Pattillo et al. (2013) for more details on the classification of cohesive clusters.

The notion of k-core is a well-studied topic with applications in disease spread (Qin et al. 2020); brain's network (Hagmann et al. 2008, Daianu et al. 2013, Shanahan et al. 2013, Wood and Hicks 2015); and social media (Malliaros and Vazirgiannis 2013). Seidman (1983) introduced the notion of k-core to serve as a way for social network researchers to measure network cohesion. Seidman (1983, p. 272) also clarifies the fact that "k-cores need not to be highly cohesive, but that all cohesive subsets are contained in k-cores." Matula and Beck (1983) showed that the maximum k-core of a graph can be computed in polynomial time. The k-core can be extended to directed graphs (Giatsidis et al. 2011b), weighted graphs (Garas et al. 2012), uncertain graphs (Bonchi et al. 2014, Peng et al. 2018), and temporal graphs (Wu et al. 2015). There are also hard, minimization, variants of k-core problem that are studied in the literature. Mikesell and Hicks (2022) use a binary integer programming model along with valid inequalities and heuristics for solving the minimum k-core problem. Ma et al. (2016) introduced the minimum spanning k-core problem with bounded probabilistic edge failures.

Bhawalkar et al. (2015) introduced the anchored k-core problem and showed that it is NP-hard for any $k \geq 3$. They propose a polynomial time algorithm to solve the anchored k-core problem when $k = 2$. Onion-layer based anchored k-core (OLAK) and residual core maximization (RCM) are two heuristic algorithms to find feasible solutions for the maximum anchored k-core problem that were proposed by Zhang et al. (2017) and Laishram et al. (2020), respectively. Tootoonchi et al. (2017) developed and implemented an efficient algorithm to solve the anchored two-core problem. Zhou et al. (2019) introduced a variant of the maximum anchored k-core problem in which a budget is spent on adding edges to the graph instead of anchoring vertices. Dey et al. (2020) studied a variant of the problem in which the budget is spent on deleting vertices and the objective is to minimize the size of the initial k-core.

3. Preliminaries

Let $G = (V, E)$ be a simple graph with vertex set V and edge set E. For every subset of vertices $S \subseteq V$, let $G[S]$ be the subgraph induced by vertex set S. For every vertex $v \in V$, we define $\deg_G(v)$ as the degree of vertex v in graph G.

When G is not specified, we denote $\deg_G(v)$ by $\deg(v)$. For every vertex $v \in V$, we define $N_G(v)$ as the open neighborhood of vertex v. We also define $n := |V|$ and $m := |E|$ as the number of vertices and edges of graph $G = (V, E)$, respectively. Now we provide some formal definitions that are used throughout the paper. We first provide a definition of the k-core as follows.

Definition 1 (k-core; Seidman 1983). The k-core of a graph $G = (V, E)$ is the maximal subset $K \subseteq V$ of vertices with $\deg_{G[K]}(v) \geq k$ for every vertex $v \in K$.

A definition of an anchored k-core is provided below.

Definition 2 (Anchored k-Core; Bhawalkar et al. 2015). Let $(C, A) \subseteq V \times V$ be an ordered set. (C, A) is an anchored k-core of graph G if and only if $\deg_{G[C \cup A]}(v) \geq k$ for every vertex $v \in C$.

Now we formally define the maximum anchored k-core problem as follows.

Problem: The maximum anchored k-core problem

Input: An undirected simple graph $G = (V, E)$ and integers k and b

Output: (if any exist) An ordered set $(C, A) \subseteq V \times V$ with a largest size of C such that $\deg_{G[C \cup A]}(v) \geq k$ for every vertex $v \in C$ and $|A| \leq b$

One can easily propose a "naïve" integer programming formulation for the maximum anchored k-core problem. For every vertex $v \in V$, binary decision variable x_v is one if vertex v belongs to a k-core set C (i.e., $\deg_{G[C \cup A]}(v) \geq k$). Furthermore, binary decision variable y_v is one if vertex v is selected as an anchor vertex (i.e., $v \in A$).

$$\text{(Naïve)} \quad \max \sum_{v \in V} x_v \tag{1a}$$

$$\sum_{u \in N_G(v)} (x_u + y_u) \geq k x_v \quad \forall v \in V, \tag{1b}$$

$$x_v + y_v \leq 1 \quad \forall v \in V, \tag{1c}$$

$$\sum_{v \in V} y_v \leq b, \tag{1d}$$

$$x, y \in \{0,1\}^n. \tag{1e}$$

Here, Objective Function (1a) maximizes the size of the anchored k-core set C. Constraints (1b) imply that if a vertex is selected in an anchored k-core set C, then at least k of its neighbors must belong to either k-core set C or anchor set A. Constraints (1c) imply that a vertex cannot belong to a k-core set C and an anchor set A simultaneously. Constraint (1d) implies that the size of an anchor set A cannot exceed budget b. Furthermore, we define the polytope of the LP relaxation of naïve Model (1) as follows:

$$\mathcal{P}_{\text{Naïve}} := \{(x, y) \in \mathbb{R}^{2n}_+ \,|\, (x, y) \text{ satisfies Constraints (1b)–(1d)}\}.$$

Because we propose multiple supervalid inequalities throughout this paper, we provide a formal definition of it as follows.

Definition 3 (Supervalid Inequality; Israeli and Wood 2002). Given polyhedron P, decision vector $x \in \mathbb{R}^n$, coefficient vectors $a, c \in \mathbb{R}^n$, and $\tau \in \mathbb{R}$ with $\arg\max_{x \in \mathbb{R}^n}\{c^T x \,|\, x \in P\} \neq \emptyset$, we say that inequality $a^T x \leq \tau$ is supervalid for P with respect to c *if*

$$\arg\max_{x \in \mathbb{R}^n}\{c^T x \,|\, x \in P\} \cap \arg\max_{x \in \mathbb{R}^n}\{c^T x \,|\, x \in P, a^T x \leq \tau\} \neq \emptyset.$$

4. Reduced IP Formulation

In this section, we propose a reduced model that is obtained by fixing a considerable number of decision variables in the naïve formulation (1). We first provide two fixing procedures before introducing the reduced model.

Remark 1 (Folklore). For every vertex $v \in V$ with $\deg_G(v) < k$, inequality $x_v \leq 0$ is valid.

Remark 1 follows by the fact that if a vertex has less than k neighbors, then the vertex cannot join any k-core set.

Proposition 1. *Let K be the k-core of graph G. For any optimal solution (x^*, y^*) of the anchored k-core problem, we have $x^*_v = 1$ and $y^*_v = 0$ for every vertex $v \in K$.*

Proof of Proposition 1. Let $(\hat{x}, \hat{y})$ be an optimal solution of the anchored k-core problem. By the contradiction, suppose that there is a vertex $v \in K$ with $\hat{x}_v = 0$. We define solution (x^*, y^*) as follows: (i) $x^*_u := \hat{x}_u$ and $y^*_u := \hat{y}_u$ for every vertex $u \in V \setminus K$, (ii) $x^*_i := 1$ for every vertex $i \in K$, and (iii) $y^*_i := 0$ for every vertex $i \in K$. By construction of

the solution (x^*, y^*), it is a feasible solution whose objective value is strictly greater than the objective value of solution $(\hat{x}, \hat{y})$. This contradicts the optimality of $(\hat{x}, \hat{y})$. □

By Remark 1 and Proposition 1, we propose a reduced IP formulation for solving the maximum anchored k-core problem. We recall that K denotes the k-core of graph G. We define the rest of vertices as $R := V \setminus K$. For every vertex $v \in R$, we define weight $w_v := |N(v) \cap K|$.

$$\text{(Reduced)} \quad \begin{aligned}
&\max |K| + \sum_{v \in R} x_v && && (2a)\\
&\sum_{u \in N_G(v) \cap R} (x_u + y_u) \geq (k - w_v) x_v && \forall v \in R \text{ with } \deg(v) \geq k, && (2b)\\
&x_v + y_v \leq 1 && \forall v \in R \text{ with } \deg(v) \geq k, && (2c)\\
&\sum_{v \in R} y_v \leq b, && && (2d)\\
&x_v = 0 && \forall v \in R \text{ with } \deg(v) < k, && (2e)\\
&x_v, y_v \in \{0,1\} && \forall v \in R. && (2f)
\end{aligned}$$

Here, Constraints (2b) imply that if a vertex $v \in R$ with w_v neighbors in the k-core set K is selected, then at least $k - w_v$ of its neighbors in R must be selected. Constraints (2c) imply that every vertex $v \in R$ with $\deg(v) \geq k$ cannot be included in both a k-core set and an anchor set simultaneously. Constraint (2d) imply that at most b vertices can be anchored. Constraints (2e) imply that by Remark 1, no vertex with a degree of less than k can be selected in a k-core set C. The reduced model cuts off some feasible solutions; however, there always exist at least one optimal solution that dominate the removed feasible ones (see Proposition 1). For analysis purposes, we rewrite the reduced IP model (2) with decision variables $x, y \in \{0,1\}^n$ as follows:

$$\begin{aligned}
&\max |K| + \sum_{v \in R} x_v, && && (3a)\\
&\sum_{u \in N_G(v) \cap R} (x_u + y_u) \geq (k - w_v) x_v && \forall v \in R \text{ with } \deg(v) \geq k, && (3b)\\
&\sum_{u \in N_G(v)} (x_u + y_u) \geq k x_v && \forall v \in K, && (3c)\\
&x_v + y_v \leq 1 && \forall v \in R \text{ with } \deg(v) \geq k, && (3d)\\
&\sum_{v \in R} y_v \leq b, && && (3e)\\
&x_v = 1 && \forall v \in K, && (3f)\\
&y_v = 0 && \forall v \in K, && (3g)\\
&x_v = 0 && \forall v \in V \text{ with } \deg(v) < k, && (3h)\\
&x_v, y_v \in \{0,1\} && \forall v \in V. && (3i)
\end{aligned}$$

Furthermore, we define the polytope of the LP relaxation of the reduced formulation (3) as follows:

$$\mathcal{P}_{\text{Reduced}} := \{(x,y) \in \mathbb{R}^{2n}_+ \mid (x,y) \text{ satisfies Constraints (3b)–(3h)}\}.$$

The following theorem shows that the LP relaxation of the reduced model (2) is at least as strong as that of the naïve formulation (1).

Theorem 1. *For every instance of the maximum anchored k-core problem, we have* $\mathcal{P}_{\text{Reduced}} \subseteq \mathcal{P}_{\text{Naïve}}$. *There exist instances for which the inclusion holds strictly.*

Proof of Theoroem 1. Consider a point $(\hat{x}, \hat{y}) \in \mathcal{P}_{\text{Reduced}}$. We are to show that $(\hat{x}, \hat{y}) \in \mathcal{P}_{\text{Naïve}}$. It suffices to show that $(\hat{x}, \hat{y})$ satisfies Constraints (1b). For every vertex $v \in R$ with $\deg(v) \geq k$, we have

$$\begin{aligned}
\sum_{u \in N_G(v)} (\hat{x}_u + \hat{y}_u) &= \sum_{u \in N_G(v) \cap K} (\hat{x}_u + \hat{y}_u) + \sum_{u \in N_G(v) \cap R} (\hat{x}_u + \hat{y}_u) && (4a)\\
&= \sum_{u \in N_G(v) \cap K} (1 + 0) + \sum_{u \in N_G(v) \cap R} (\hat{x}_u + \hat{y}_u) && (4b)
\end{aligned}$$

$$= w_v + \sum_{u \in N_G(v) \cap R} (\hat{x}_u + \hat{y}_u) \tag{4c}$$

$$\geq w_v \hat{x}_v + \sum_{u \in N_G(v) \cap R} (\hat{x}_u + \hat{y}_u) \tag{4d}$$

$$\geq w_v \hat{x}_v + (k - w_v)\hat{x}_v \tag{4e}$$

$$= k\hat{x}_v. \tag{4f}$$

Here, Equality (4b) holds by Constraints (3f) and (3g). Equality (4c) holds by the definition of w. Inequality (4d) holds because $\hat{x} \in [0,1]^n$. Finally, Inequality (4e) holds by Constraints (3b).

For every vertex $v \in R$ with $\deg(v) < k$, it is easy to see that $(\hat{x}, \hat{y})$ satisfies Constraints (1b) as $\hat{x}_v = 0$ for every vertex $v \in R$ with $\deg(v) < k$ by Constraints (3h). Furthermore, for every vertex $v \in K$, we have

$$\sum_{u \in N_G(v)} (\hat{x}_u + \hat{y}_u) \geq \sum_{u \in N_G(v) \cap K} (\hat{x}_u + \hat{y}_u) \geq k = k\hat{x}_v.$$

Here, the first inequality holds by nonnegativity bounds of x and y variables. The second inequality holds by the fact that $v \in K$ and by Constraints (3f). The equality holds because $\hat{x}_v = 1$ by Constraints (3f). Finally, the following example shows that the inclusion can be strict.

Example 1. Figure 2 provides a point $(\hat{x}, \hat{y}) \in \mathcal{P}_{\text{Naïve}}$ such that $(\hat{x}, \hat{y}) \notin \mathcal{P}_{\text{Reduced}}$. □

5. Polyhedral Study

In this section, we conduct a polyhedral study on the polytope of the maximum anchored k-core problem in a reduced space. Without loss of generality, we assume that all vertices of G are labeled from 1 to n. We first define set R' as follows:

$$R' = \{u \in R | \deg(u) \geq k\}.$$

We recall that $R = V \setminus K$, where K is the set of the k-core of graph G. We set $r := |R|$ and $r' := |R'|$ and define the polytope of the maximum anchored k-core problem as follows:

$$P_{k,b}(G) := \text{conv}\ \{(x^Q, y^A) \in \{0,1\}^{r'+r} | (K \cup Q, A) \text{ forms an anchored } k\text{-core with} |A| \leq b\},$$

where x^Q and y^A are the characteristic vectors of $Q \subseteq R'$ and $A \subseteq R$, respectively. Throughout this section, we use e_i to denote the unit vector of appropriate size corresponding to vertex $i \in V$. We also introduce two points in Definitions 4 and 5 that are used in the proofs of this section.

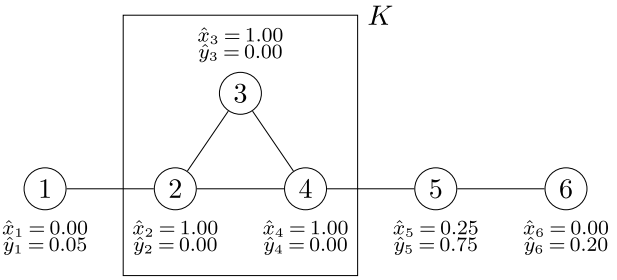
Definition 4. Let $b \geq k$. For any vertex $u \in R'$, we define $q_u \in \{0,1\}^{r'+r}$ as a binary vector that represents a solution in which

i. Only vertex $u \in R'$ is selected in a k-core; and
ii. Exactly k neighbors of vertex u are anchored.

Definition 5. Suppose $b \geq 2k - 2$ and let vertex $u \in R'$. For every vertex $j \in N_G(u) \cap R'$, we define $h_{u,j} \in \{0,1\}^{r'+r}$ as a binary vector that represents a solution in which

i. Vertices u and j are selected in a k-core;
ii. Exactly $k - 1$ neighbors of vertex u, excluding vertex j, are anchored; and
iii. Exactly $k - 1$ neighbors of vertex j, excluding vertex u, are anchored.

Figure 2. Instance of the Anchored k-Core Problem with $k = 2$, $b = 1$, $K = \{2, 3, 4\}$, and $R = \{1, 5, 6\}$



Note. Although $(\hat{x}, \hat{y}) \in \mathcal{P}_{\text{Naïve}}$, it violates Constraints (3b) of Formulation (3) for vertex 5; that is, $0.20 = \hat{x}_6 + \hat{y}_6 = \sum_{j \in N_G(5) \cap R} (\hat{x}_j + \hat{y}_j) \not\geq (k - w_5)\hat{x}_5 = \hat{x}_5 = 0.25$.

The following proposition shows that polytope $P_{k,b}(G)$ is full-dimensional under a reasonable condition (i.e., $b \geq k$).[3]

Proposition 2. *Polytope $P_{k,b}(G)$ is full-dimensional if and only if $b \geq k$.*

Proof of Proposition 2. ($\Rightarrow$) By the contrapositive. Suppose that $b \leq k-1$. Then, Example 2 shows that there is an instance of the maximum anchored k-core problem for which the $P_{k,b}(G)$ polytope is not full-dimensional.

Example 2. Consider an instance of the maximum anchored k-core problem with $k=3$ and $b=2$ shown in Figure 3.

A minimal description of $P_{3,2}(G)$ is provided below using PORTA (Christof and Loebel 2022).

$$
\begin{array}{rrrrrr}
+x_2 & -x_3 & & & & = 0 \\
 & & & & & \\
 & -x_3 & & & & \leq 0 \\
 & & -y_2 & & & \leq 0 \\
 & & & -y_3 & & \leq 0 \\
+x_3 & -y_1 & & & & \leq 0 \\
+x_3 & & & & -y_4 & \leq 0 \\
 & & & & +y_4 & \leq 1 \\
 & +y_1 & & & & \leq 1 \\
+x_3 & & & +y_3 & & \leq 1 \\
+x_3 & & +y_2 & & & \leq 1 \\
 & +y_1 & +y_2 & +y_3 & +y_4 & \leq 2
\end{array}
$$

By equality $x_2 - x_3 = 0$, we do not have a unique description for $P_{3,2}(G)$. Furthermore, the description is minimal. Hence, $P_{3,2}(G)$ is not full-dimensional by corollary 3.31 of Conforti et al. (2014).

($\Leftarrow$) First, $(\mathbf{0},\mathbf{0})^T \in P_{k,b}(G)$. We also have $(\mathbf{0},e_u)^T \in P_{k,b}(G)$ for every vertex $u \in R$. We now introduce r' points as follows. For every vertex $u \in R'$, we define point q_u by Definition 4. Therefore, we have the following $r'+r$ linearly independent points: $(\mathbf{0},e_1)^T - (\mathbf{0},\mathbf{0})^T, (\mathbf{0},e_2)^T - (\mathbf{0},\mathbf{0})^T, \ldots, (\mathbf{0},e_r)^T - (\mathbf{0},\mathbf{0})^T$, and $q_u - \mathbf{0}$ for all u $\in R'$. Hence, we have $r'+r+1$ affinely independent vectors in polyhedron $P_{k,b}(G)$. □

Now we show multiple inequalities of Formulation (2) are facet-defining under mild conditions. The following proposition shows that the nonnegativity bounds on x variables induce facets of $P_{k,b}(G)$ if $b \geq k$.

Proposition 3. *If $b \geq k$, then $x_u \geq 0$ is facet-defining for every vertex $u \in R'$.*

Proof of Proposition 3. First, point $(\mathbf{0},\mathbf{0})^T \in P_{k,b}(G)$ satisfies the inequality at equality. For every vertex $u \in R$, we define $(\mathbf{0},e_u)^T \in P_{k,b}(G)$. By Definition 4, we define point q_v for every vertex $v \in R' \setminus \{u\}$. Therefore, we have $r'+r$ affinely independent points. This finishes the proof. □

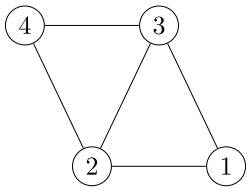
The following proposition shows that conflict Constraints (2c) are facet-defining if $b \geq k+1$.

Proposition 4. *The expression $x_u + y_u \leq 1$ is facet-defining for every vertex $u \in R'$ if and only if $b \geq k+1$.*

Proof of Proposition 4. ($\Rightarrow$) By the contrapositive. Suppose that $b \leq k$. Then there is an instance of the maximum anchored k-core problem for which the inequality is not facet-defining. The instance is provided in Online Appendix A.

($\Leftarrow$) We start with defining r points that satisfy the inequality at equality; that is, $(\mathbf{0},e_u)^T$ and $(\mathbf{0},e_u+e_i)^T$ for every vertex $i \in R \setminus \{u\}$. For every vertex $v \in R' \setminus \{u\}$, we define $\overline{q}_v$ such that element $\overline{q}_v^i = q_v^i$ for every index $i \in$

Figure 3. Instance of the Maximum Anchored k-Core Problem with $k=3$ and $b=2$



$\{1,2,\ldots,r'+r\}\setminus\{r'+u\}$ and $\overline{q}_v^{r'+u}=1$. Along with q_u, we have r' more points. We summarize $r'+r$ affinely independent points that satisfy the inequality at equality as follows.

i. $(\mathbf{0},e_u+e_i)^T$ for every vertex $i\in R\setminus\{u\}$,
ii. $(\mathbf{0},e_u)^T$,
iii. $\overline{q}_v$ for every vertex $v\in R'\setminus\{u\}$, and
iv. q_u.

Figure 4 presents $r'+r$ affinely independent points that satisfy Constraints (2c) at equality. □

We now prove that inequality $y_u\le 1$ is facet-defining for every vertex $u\in R\setminus R'$ when $b\ge k+1$.

Proposition 5. *The expression $y_u\le 1$ is facet-defining for every vertex $u\in R\setminus R'$ if and only if $b\ge k+1$.*

Proof of Proposition 5. ($\Rightarrow$) By the contrapositive. Suppose that $b\le k$. Then there is an instance of the maximum anchored k-core problem for which the inequality is not facet-defining. The instance is provided in Online Appendix A.

($\Leftarrow$) We first define r points as follows: $(\mathbf{0},e_u)$ and $(\mathbf{0},e_i+e_u)$ for every vertex $i\in R\setminus\{u\}$. Now we define r' points. For every vertex $v\in R'$, consider point q_v defined in Definition 4. For every $v\in R'$, we define point $\overline{q}_v$ with elements $\overline{q}_v^i=q_v^i$ for every index $i\in\{1,2,\ldots,r'+r\}\setminus\{r'+u\}$ and $\overline{q}_v^{r'+u}=1$. Now, we have $r'+r$ affinely independent points that are summarized as follows.

i. $(\mathbf{0},e_u)^T$,
ii. $(\mathbf{0},e_u+e_i)^T$ for every vertex $i\in R\setminus\{u\}$, and
iii. $\overline{q}_v$ for every vertex $v\in R'$.

Figure 5 shows $r'+r$ affinely independent points that satisfy inequality $y_u\le 1$ at equality for every vertex $u\in R\setminus R'$. □

The following proposition shows that nonnegativity bounds on y variables of the vertex set R' are facet-defining if $b\ge 2k-2$.

Proposition 6. *The expression $y_u\ge 0$ is facet-defining for every vertex $u\in R'$ if and only if $b\ge 2k-2$.*

Proof of Proposition 6. ($\Rightarrow$) By the contrapositive. We assume that $b\le 2k-3$. Then, there is an instance of the maximum anchored k-core problem for which the inequality is not facet-defining. The instance is provided in Online Appendix A.

($\Leftarrow$) First we consider point $(\mathbf{0},\mathbf{0})$. We also define $r-1$ points $(\mathbf{0},e_i)$ for every vertex $i\in R\setminus\{u\}$. Based on Definitions 4 and 5, we construct r' points as follows: (i) q_j for every vertex $j\in R'\setminus N_G(u)$, and (ii) $h_{u,j}$ for every vertex $j\in N_G(u)\cap R'$. Figure 6 presents $r'+r$ affinely independent points that satisfy the inequality at equality. We also define $\alpha:=|R'\setminus N_G(u)|$. Without loss of generality, we label (i) vertices of the set $R'\setminus N_G(u)$ from 1 to α, and (ii) vertices of the set $R'\cap N_G(u)$ from $\alpha+1$ to r'. □

Next proposition shows under what conditions the budget constraint (2d) is facet-defining.

Proposition 7. *If $k\le b\le r-1$, then the budget constraint (2d) is facet-defining for $P_{k,b}(G)$.*

Figure 4. Collection of $r'+r$ Affinely Independent Points Satisfying Constraints (2c) at Equality

	1	2	$\cdots$	u	$\cdots$	$r'-1$	r'	$r'+1$	$r'+2$	$\cdots$	$r'+u$	$\cdots$	$r'+r-1$	$r'+r$
1 2 $\vdots$ u $\vdots$ r'	**I**							**0**						
$r'+1$	$q_1^{r'+1}$	$q_2^{r'+1}$	$\cdots$	$q_u^{r'+1}$	$\cdots$	$q_{r'-1}^{r'+1}$	$q_{r'}^{r'+1}$	1	0	0	0	0	0	0
$r'+2$	$q_1^{r'+2}$	$q_2^{r'+2}$	$\cdots$	$q_u^{r'+2}$	$\cdots$	$q_{r'-1}^{r'+2}$	$q_{r'}^{r'+2}$	0	1	0	0	0	0	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$			$\ddots$				
$r'+u$	1	1	$\cdots$	0	$\cdots$	1	1	1	1	$\cdots$	1	$\cdots$	1	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$					$\ddots$		
$r'+r-1$	$q_1^{r'+r-1}$	$q_2^{r'+r-1}$	$\cdots$	$q_u^{r'+r-1}$	$\cdots$	$q_{r'-1}^{r'+r-1}$	$q_{r'}^{r'+r-1}$	0	0	0	0	0	1	0
$r'+r$	$q_1^{r'+r}$	$q_2^{r'+r}$	$\cdots$	$q_u^{r'+r}$	$\cdots$	$q_{r'-1}^{r'+r}$	$q_{r'}^{r'+r}$	0	0	0	0	0	0	1

Figure 5. Collection of $r' + r$ Affinely Independent Points Satisfying Inequality $y_u \leq 1$ at Equality

	1	2	$\cdots$	$r'-1$	r'	$r'+1$	$r'+2$	$\cdots$	$r'+r-1$	$r'+r$
1										
2										
$\vdots$			**I**					**0**		
r'										
$r'+1$	$q_1^{r'+1}$	$q_2^{r'+1}$	$\cdots$	$q_{r'-1}^{r'+1}$	$q_{r'}^{r'+1}$	1	0	$\cdots$	0	0
$r'+2$	$q_1^{r'+2}$	$q_2^{r'+2}$	$\cdots$	$q_{r'-1}^{r'+2}$	$q_{r'}^{r'+2}$	0	1	$\cdots$	0	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$r'+u$	1	1	$\cdots$	1	1	1	1	$\cdots$	1	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$r'+r-1$	$q_1^{r'+r-1}$	$q_2^{r'+r-1}$	$\cdots$	$q_{r'-1}^{r'+r-1}$	$q_{r'}^{r'+r-1}$	0	0	$\cdots$	1	0
$r'+r$	$q_1^{r'+r}$	$q_2^{r'+r}$	$\cdots$	$q_{r'-1}^{r'+r}$	$q_{r'}^{r'+r}$	0	0	$\cdots$	0	1

Proof of Proposition 7. Figure 7 shows $r' + r$ affinely independent points that satisfy Inequality (2d) at equality. The summation of the bottom part of each column of matrix W equals b (see caption of Figure 7 for a description of submatrix $\overline{Q}_{r\times r'}$). Now, we show that these points are affinely independent. Let $c_1, c_2, \ldots, c_{r'+r}$ be the columns of matrix W in Figure 7. We are to show that

$$\sum_{i=1}^{r'+r} \lambda_i c_i = 0, \quad \text{and} \sum_{i=1}^{r'+r} \lambda_i = 0, \tag{5}$$

imply $\lambda_j = 0$ for every $j \in \{r+1, r+2, \ldots, r'+r\}$. By the first equality of Line (5) for the top-right of matrix W, we have $\lambda_{r+1} = \lambda_{r+2} = \cdots = \lambda_{r'+r} = 0$ because columns of the identity submatrix $I_{r'}$ are linearly independent. Columns of the bottom-left submatrix of W are linearly independent because the submatrix is the transpose of the nonsingular matrix M in the proof of corollary 4.4 in Nemhauser and Trotter (1974). We note that k and t of their paper are defined as $k := b + 1$ and $t := r$ with b and r in our paper. □

Figure 6. Collection of $r' + r$ Affinely Independent Points Satisfying Constraints $y_u \geq 0$ at Equality for Every Vertex $u \in$ R′

	1	2	$\ldots$	α	$\alpha+1$	$\alpha+2$	$\ldots$	r'	$r'+u$	$r'+1$	$r'+2$	$\cdots$	$r'+u-1$	$r'+u+1$	$\cdots$	$r'+r$
1					0	0	0	0	0							
2					0	0	0	0	0							
$\vdots$					$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$							
$u-1$			**I**		0	0	0	0	0			**0**				
u					1	1	1	1	0							
$u+1$					0	0	0	0	0							
$\vdots$					$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$							
α					0	0	0	0	0							
$\alpha+1$									0							
$\alpha+2$									0							
$\vdots$			**0**				**I**		$\vdots$			**0**				
r'									0							
$r'+u$	0	0	$\cdots$	0	0	0	$\cdots$	0	0	0	0	$\cdots$	0	0	$\cdots$	0
$r'+1$	$q_1^{r'+1}$	$q_2^{r'+1}$	$\cdots$	$q_\alpha^{r'+1}$	$h_{u,\alpha+1}^{r'+1}$	$h_{u,\alpha+2}^{r'+1}$	$\cdots$	$h_{u,r'}^{r'+1}$	0							
$r'+2$	$q_1^{r'+2}$	$q_2^{r'+2}$	$\cdots$	$q_\alpha^{r'+2}$	$h_{u,\alpha+1}^{r'+2}$	$h_{u,\alpha+2}^{r'+2}$	$\cdots$	$h_{u,r'}^{r'+2}$	0							
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$							
$r'+u-1$	$q_1^{r'+u-1}$	$q_2^{r'+u-1}$	$\cdots$	$q_\alpha^{r'+u-1}$	$h_{u,\alpha+1}^{r'+u-1}$	$h_{u,\alpha+2}^{r'+u-1}$	$\cdots$	$h_{u,r'}^{r'+u-1}$	0				**I**			
$r'+u+1$	$q_1^{r'+u+1}$	$q_2^{r'+u+1}$	$\cdots$	$q_\alpha^{r'+u+1}$	$h_{u,\alpha+1}^{r'+u+1}$	$h_{u,\alpha+2}^{r'+u+1}$	$\cdots$	$h_{u,r'}^{r'+u+1}$	0							
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$							
$r'+r$	$q_1^{r'+r}$	$q_2^{r'+r}$	$\cdots$	$q_\alpha^{r'+r}$	$h_{u,\alpha+1}^{r'+r}$	$h_{u,\alpha+2}^{r'+r}$	$\cdots$	$h_{u,r'}^{r'+r}$	0							

Figure 7. $r' + r$ Affinely Independent Points That Satisfy Inequality (2d) at Equality

$$W = \left[\begin{array}{c|c|c} \multicolumn{2}{c|}{\mathbf{0}_{r'\times r}} & \mathbf{I}_{r'} \\ \hline (\mathbf{1}-\mathbf{I})_{b+1} & L_{(b+1)\times(r-(b+1))} & \multirow{2}{*}{$\bar{Q}_{r\times r'}$} \\ \hline \mathbf{0}_{(r-(b+1))\times(b+1)} & \mathbf{I}_{r-(b+1)} & \end{array}\right]$$

Notes. Here, L is a matrix with $r-(b+1)$ columns of form $(1,\ldots,1,0,0)^T_{b+1}$. Matrices 1 and 0 represent matrices with all one and all zero entities, respectively. Square matrix I represents the identity matrix. Further, $\bar{Q}$ represents vectors $\bar{q}_1, \bar{q}_2, \ldots, \bar{q}_{r'}$ where $\bar{q}_i \in \{0,1\}^r$ is the subvector of $q_i \in \{0,1\}^{r'+r}$ with $\bar{q}_i^j := q_i^{r'+j}$ for all $i \in \{1, 2, \ldots, r'\}$ and $j \in \{1, 2, \ldots, r\}$.

6. More Inequalities and Fixings

In this section, we propose valid and supervalid inequalities and fixing procedures to strengthen our reduced model. In Section 7, we will test the efficiency of these inequalities and fixing procedures computationally. Proposition 8 proposes a new set of valid inequalities and the condition under which the inequalities are facet-defining.

Proposition 8. *Let $v \in R'$ be a vertex with $\deg(v) = k$. Then $x_v \le x_u + y_u$ is* (i) *valid and* (ii) *facet-defining for every vertex $u \in N_G(v) \cap R'$ if and only if $b \ge 2k-2$.*

Proof of Proposition 8. ($\Rightarrow$) By the contrapositive. Suppose that $b \le 2k-3$. Then, there is an instance of the maximum anchored k-core problem for which the inequality is not facet-defining. The instance is provided in Online Appendix A.

($\Leftarrow$) First, we show that the inequality is valid. Let (C, A) be an anchored k-core for graph G, and let $v \in R'$ be a vertex with $\deg(v) = k$ and vertex $u \in N_v(G)$. Furthermore, let $(\hat{x}, \hat{y})$ be the point corresponding to (C, A). If $v \notin C$, then the inequality holds trivially because $\hat{x}_v = 0$. Now, suppose that $v \in C$. Then all neighbors of vertex v must belong to $C \cup A$. This means that $(\hat{x}, \hat{y})$ satisfies the inequality for every vertex $u \in N_G(v)$.

Now, we prove the second claim. Points (i) $(\mathbf{0}, \mathbf{0})^T$ and (ii) $(\mathbf{0}, e_j)^T$ for every vertex $j \in R \setminus \{u\}$ satisfy the inequality at equality. Based on Definitions 4 and 5, we construct r' points as follows: (i) q_j for every vertex $j \in R' \setminus N_G(v)$ and (ii) $h_{v,j}$ for every vertex $j \in N_G(v) \cap R'$. We can see that the points are affinely independent by the linear independence of all nonzero points subtracted by point $(\mathbf{0}, \mathbf{0})^T$.

Figure 8 presents $r' + r$ affinely independent points that satisfy constraints $x_v \le x_u + y_u$ for vertex $v \in R'$ with $\deg(v) = k$ and a vertex $u \in N_G(v) \cap R'$. Without loss of generality, we assume that (i) $u = v + 1$ and (ii) $v, u, u+1, \ldots, u+k-1$ represent labels of all neighbors of vertex v. □

Proposition 9 proposes a set of supervalid inequalities (i.e., inequalities that might cut off some integer feasible solutions, but at least one optimal solution remains (see Definition 3)).

Proposition 9. *For any vertex $v \in V$ and vertex $u \in V$, suppose that*

- $\deg(u) < k$, *and*
- $N_G(u) \setminus \{v\} \subset N_G(v) \setminus \{u\}$.

Then $x_v + y_v \ge y_u$ is a supervalid inequality with respect to Objective Function (1a).

Figure 8. Collection of $r' + r$ Affinely Independent Points Satisfying Constraint $x_v \le x_u + y_u$ at Equality

	1	$\cdots$	$v-1$	v	u	$u+1$	$\cdots$	$u+k-1$	$u+k$	$u+k+1$	$\cdots$	r'	$r'+1$	$\cdots$	$r'+u$	$\cdots$	$r'+r$
1				0	0	0	0	0									
2				0	0	0	0	0									
$\vdots$				$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$									
$v-1$		**I**		0	0	0	0	0		**I**				**0**			
v				1	1	1	1	1									
u				0	1	0	0	0									
$u+1$				0	0	1	0	0									
$\vdots$				$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$									
r'				0	0	0	0	1									
$r'+1$	$q_1^{r'+1}$	$\cdots$	$q_{v-1}^{r'+1}$	$q_v^{r'+1}$	$h_{v,u}^{r'+1}$	$h_{v,u+1}^{r'+1}$	$\cdots$	$h_{v,u+k-1}^{r'+1}$	$q_{u+k}^{r'+1}$	$q_{u+k+1}^{r'+1}$	$\cdots$	$q_{r'}^{r'+1}$	1	$\cdots$	0	$\cdots$	0
$r'+2$	$q_1^{r'+2}$	$\cdots$	$q_{v-1}^{r'+2}$	$q_v^{r'+2}$	$h_{v,u}^{r'+2}$	$h_{v,u+1}^{r'+2}$	$\cdots$	$h_{v,u+k-1}^{r'+2}$	$q_{u+k}^{r'+2}$	$q_{u+k+1}^{r'+2}$	$\cdots$	$q_{r'}^{r'+2}$	0	$\cdots$	0	$\cdots$	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$r'+u$	$q_1^{r'+u}$	$\cdots$	$q_{v-1}^{r'+u}$	$q_v^{r'+u}$	$h_{v,u}^{r'+u}$	$h_{v,u+1}^{r'+u}$	$\cdots$	$h_{v,u+k-1}^{r'+u}$	$q_{u+k}^{r'+u}$	$q_{u+k+1}^{r'+u}$	$\cdots$	$q_{r'}^{r'+u}$	0	$\cdots$	0	$\cdots$	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$r'+r-1$	$q_1^{r'+r-1}$	$\cdots$	$q_{v-1}^{r'+r-1}$	$q_v^{r'+r-1}$	$h_{v,u}^{r'+r-1}$	$h_{v,u+1}^{r'+r-1}$	$\cdots$	$h_{v,u+k-1}^{r'+r-1}$	q_{u+k}^{r+r-1}	q_{u+k+1}^{r+r-1}	$\cdots$	$q_{r'}^{r+r-1}$	0	$\cdots$	0	$\cdots$	0
$r'+r$	$q_1^{r'+r}$	$\cdots$	$q_{v-1}^{r'+r}$	$q_v^{r'+r}$	$h_{v,u}^{r'+r}$	$h_{v,u+1}^{r'+r}$	$\cdots$	$h_{v,u+k-1}^{r'+r}$	$q_{u+k}^{r'+r}$	$q_{u+k+1}^{r'+r}$	$\cdots$	$q_{r'}^{r'+r}$	0	$\cdots$	0	$\cdots$	1

Proof of Proposition 9. Let $(\hat{x},\hat{y})$ be an optimal solution of the anchored k-core problem. Furthermore, suppose that $\deg(u) < k$ and $N_G(u) \setminus \{v\} \subset N_G(v) \setminus \{u\}$ holds for arbitrary vertices $v \in V$ and $u \in V$. If (i) $\hat{y}_u = 0$, or (ii) $\hat{y}_u = 1$ and $\hat{x}_v = 1$, or (iii) $\hat{y}_u = 1$ and $\hat{y}_v = 1$, then $\hat{x}_v + \hat{y}_v \geq \hat{y}_u$ holds, and we are done. Now suppose that $\hat{y}_u = 1$ and $\hat{x}_v = 0$ and $\hat{y}_v = 0$. We define solution (x^*,y^*) as follows: (i) $x_i^* = \hat{x}_i$ for all $i \in V$, (ii) $y_i^* = \hat{y}_i$ for all $i \in V \setminus \{u,v\}$, and (iii) $y_u^* = 0$ and $y_v^* = 1$. As $x^* = \hat{x}$, the objective values corresponding to points (x^*,y^*) and $(\hat{x},\hat{y})$ are equivalent. Thus, (x^*,y^*) is optimal. Now we show that point (x^*,y^*) is a feasible solution. The point satisfies degree Constraints (1b) as $N_G(u) \setminus \{v\} \subset N_G(v) \setminus \{u\}$ and $\deg(u) < k$. The conflict constraints (1c) are satisfied by construction. We finally show that it satisfies the budget constraint (1d) as follows:

$$\sum_{i\in V} y_i^* = y_u^* + y_v^* + \sum_{i\in V\setminus\{u,v\}} y_i^* = y_u^* + y_v^* + \sum_{i\in V\setminus\{u,v\}} \hat{y}_i = \hat{y}_v + \hat{y}_u + \sum_{i\in V\setminus\{u,v\}} \hat{y}_i = \sum_{i\in V} \hat{y}_i \leq b.$$

This concludes the proof. □

Proposition 10 provides a fixing procedure for y variables.

Proposition 10. *Let $v \in V$ be a vertex with $\deg(u) < k$ for every vertex $u \in N(v)$. Then there exists an optimal solution (x^*,y^*) with $y_v^* = 0$.*

Proof of Proposition 10. Let $(\hat{x},\hat{y})$ be an optimal solution of the anchored k-core problem. If $\hat{y}_v = 0$, then we define $x^* := \hat{x}$ and $y^* := \hat{y}$, and we are done. Now suppose that $\hat{y}_v = 1$. We define solution (x^*,y^*) as follows: (i) $x^* = \hat{x}$, (ii) $y_i^* = \hat{y}_i$ for every vertex $i \in V \setminus \{v\}$, and (iii) $y_v^* = 0$. As $x^* = \hat{x}$, the objective values corresponding to points (x^*,y^*) and $(\hat{x},\hat{y})$ are equivalent. Now we show that point (x^*,y^*) is also a feasible solution. The point satisfies the degree constraints (1b) and the conflict constraints (1c) by Remark 1. It suffices to show that the point satisfies the budget constraint (1d) as follows:

$$\sum_{i\in V} y_i^* = y_v^* + \sum_{i\in V\setminus\{v\}} y_i^* = y_v^* + \sum_{i\in V\setminus\{v\}} \hat{y}_i < \hat{y}_v + \sum_{i\in V\setminus\{v\}} \hat{y}_i = \sum_{i\in V} \hat{y}_i \leq b.$$

This concludes the proof. □

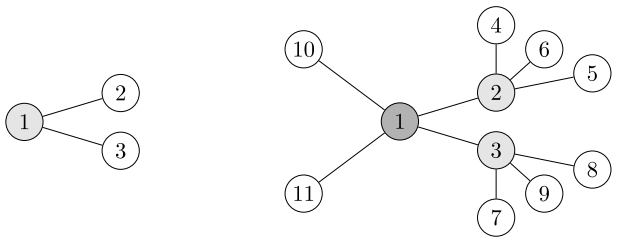
Proposition 11 proposes a fixing procedure for fixing x variables to zero when $b < k$. This is a reasonable assumption as we observe benchmark instances with $b < k$ in Zhang et al. (2017). This fixing procedure finds a set of vertices $U \subseteq V$ with degree at least k such that any solution (C,A) to the anchored k-core problem satisfies $U \cap C = \emptyset$. Figure 9 illustrates two instances of the anchored k-core problem where Proposition (11) yields fixings. On the left side of Figure 9, one can fix variable x_1 to zero for vertex 1 because we do not have enough budget to anchor (or buy) both 2 and 3 for activating vertex 1 as a vertex in a k-core. On the right side of Figure 9, we observe that the fixing can be applied iteratively to fix vertices 2, 3, and 1.

Proposition 11. *Let $v \in R'$, and $S_v = N_G(v) \cap Q$, where Q is a set of vertices for which x variables are not fixed to zero. If $|S_v| + b < k$, then inequality $x_v \leq 0$ is valid.*

Proof of Proposition 11. By the contradiction. Suppose that there exists a solution $(\hat{x},\hat{y})$ with $\hat{x}_v = 1$ for a vertex $v \in V$ with $|S_v| + b < k$. By Constraints (1b), we have

$$\sum_{u\in N_G(v)} (\hat{x}_u + \hat{y}_u) \geq k\hat{x}_v.$$

Figure 9. Fixing Procedure of Proposition 11



Notes. (Left) Variable x_1 is fixed to zero for vertex 1 in an anchored k-core instance with $k = 2$ and $b = 1$. (Right) For an anchored k-core instance with $k = 4$ and $b = 2$, (i) we first fix variables x_2 and x_3 to zero for vertices 2 and 3, respectively, and then (ii) we fix variable x_1 to zero because x_2, x_3, x_{10}, and x_{11} are fixed to zero and budget b is less than k.

By the definition of S_v, we have

$$\sum_{u \in N_G(v)} \hat{x}_u \leq |S_v|. \tag{6}$$

Furthermore, we have

$$\sum_{u \in N_G(v)} \hat{y}_u \leq \sum_{u \in V} \hat{y}_u \leq b. \tag{7}$$

Here, the first inequality holds because $\hat{y} \geq 0$. The second inequality holds by budget Constraint (1d).

By Inequalities (6) and (7), we have

$$\sum_{u \in N_G(v)} (\hat{x}_u + \hat{y}_u) \leq |S_v| + b. \tag{8}$$

This is a contradiction as

$$k = k\hat{x}_v \leq \sum_{u \in N_G(v)} (\hat{x}_u + \hat{y}_u) \leq |S_v| + b < k.$$

Here, the first equality holds because $\hat{x}_v = 1$. The first inequality holds by Constraints (1b). The second inequality holds by Inequality (8). The last inequality holds by the assumption. □

On the left side of Figure 9, we have $Q = \emptyset$ and $S_1 = \emptyset$; thus, $k = 2 > 0 + 1 = |S_1| + b$ and we can safely fix x_1 to zero. On the right side of Figure 9, we first note that $S_2 = S_3 = \{1\}$. Because $k = 4 > 1 + 2 = |S_2| + b$ and $k = 4 > 1 + 2 = |S_3| + b$, we can fix variables x_2 and x_3 to zero, respectively. Then we can fix x_1 to zero as $k = 4 > 0 + 2 = |S_1| + b$.

7. Computational Experiments

In this section, we computationally compare the performance of the reduced model (2) with the naïve formulation (1) and two existing heuristic approaches: RCM and OLAK. We also test the computational performance of the inequalities and fixing procedures proposed in Section 6. We run our experiments on two sets of benchmark instances whose details are provided in Table 1. All experiments are conducted on a machine running Red Hat Enterprise Linux Workstation x64 version 7.6 with an Intel(R) Core(TM) i7-9800X CPU (3.8 Ghz, 19.25 MB, 165 W) using one core with 32 GB RAM.

We use Python to implement our algorithms and mathematical models. We use Gurobi 9.5 as the IP solver. Furthermore, we set a time limit (TL) of 3,600 seconds for all of our computational experiments. The k-core of the graph is computed by the k_core function of Networkx in Python.[4] The implementation is borrowed from the $O(m)$ algorithm

Table 1. Benchmark Instances of RCM (Laishram et al. 2020) and OLAK (Zhang et al. 2017)

Instance	Abbreviation	src	n	m	d_{avg}	d_{max}	k_{max}	k_{med}
facebook-combined	FC	SNAP	4,039	88,234	25	1,045	115	17
CA-HepPh	HP	SNAP	12,006	118,489	5	491	238	4
socfb-Syracuse	FS	NR	13,653	543,982	62	1,340	75	46
socfb-Northeastern	FN	NR	13,882	381,934	42	968	43	33
CA-CondMat	CM	SNAP	23,133	93,439	5	279	25	4
Brightkite-edges	BK	SNAP	58,288	214,078	2	1,134	52	2
Flickr	FL	SNAP	105,938	2,316,948	7	5,425	573	5
soc-catster	CA	NR	149,684	5,448,197	22	80,634	419	21
Gowalla-edges	GW	SNAP	196,591	950,327	3	14,730	51	3
ca-citeseer	CS	NR	227,320	814,134	4	1,372	86	3
com-dblp	DB	SNAP	317,080	1,049,866	4	343	113	3
soc-Dogster	DO	NR	426,816	8,543,549	12	46,503	248	12
soc-TwitterHiggs	TH	NR	456,631	12,508,442	18	51,386	125	17
web-Google	GO	SNAP	875,713	4,322,051	5	6,332	44	4
com-Youtube	YT	SNAP	1,134,890	2,987,624	1	28,754	51	1
web-Hudong	HU	NR	1,974,655	14,428,382	5	61,440	266	5
web-BaiduBaike	BB	NR	2,140,198	17,014,946	4	97,848	78	3

Note. We report source of data sets (src), number of vertices (n), number of edges (m), average degree (d_{avg}), maximum degree (d_{max}), maximum possible k (k_{max}), and median k (k_{med}).

of Batagelj and Zaversnik (2003). The k-core computation time is included in the run time of the Reduced model. Our codes, data, and results are available at https://github.com/samuel-kroger/Maximizing-engagement-in-large-scale-social-networks.

Table 1 provides information about two sets of benchmark instances from Laishram et al. (2020) and Zhang et al. (2017). These instances are available at the Stanford Large Network Data Set Collection (SNAP) (Leskovec and Krevl 2014) and the Network Repository (NR) (Rossi and Ahmed 2015). Columns abv and src represent abbreviations of the instances and their sources, respectively. The number of vertices and the number of edges are denoted in columns n and m, respectively. Columns d_{avg} and d_{max} indicate the average degree of vertices and the maximum degree of vertices, respectively. To explain the values in columns k_{max} and k_{med}, Algorithm 1 provides a k-core decomposition that returns the coreness of each vertex (Seidman 1983). Function kcore(G', k) in Algorithm 1 returns the k-core set of graph G'.

Algorithm 1 (k-Core Decomposition $(G = (V, E))$)

coreness$(v) \leftarrow 0$ for every vertex $v \in V$
$k \leftarrow 1$
while True **do**
  $G' = (V', E') \leftarrow G = (V, E)$
  $C \leftarrow$ kcore(G', k)
  if $C = \emptyset$: **break**
  while $C \neq \emptyset$ **do**
    for $v \in C$ **do**
      coreness$(v) \leftarrow k$
      $G' \leftarrow G' - v$
    $C \leftarrow$ kcore(G', k)
  $k \leftarrow k + 1$
return coreness

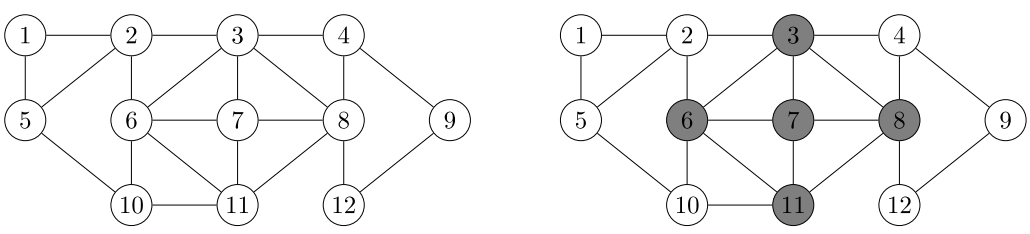
Figure 10 provides the k-core decomposition of the researchers' social network provided in the introduction. On the right side of the figure, white vertices and gray vertices have corenesses of two and three, respectively. Hence, k_{med} (i.e., the median of the coreness values) and k_{max} (i.e., the maximum of the coreness values) of the researchers' social network are two and three, respectively.

Laishram et al. (2020) set the value of k to k_{med} in their experiments. In Tables 2–8, RCM instances of Laishram et al. (2020) are provided above the horizontal line, and OLAK instances of Zhang et al. (2017) are listed below the horizontal line.

7.1. Reduced Model vs. Naïve Model

Table 2 compares the computational performance of the reduced model (2) with that of the naïve one (1). One can observe that the reduced model outperforms the naïve formulation in all but one of the instances by either time or optimality gap. Although the naïve model struggles or fails to solve the problem for CA, DO, TH, GO, HU, and BB in RCM benchmark instances (above the horizontal line), the reduced model solves all of them in the time limit. When time limit is reached for both models in FS and FN (from RCM instances) and GW and YT (from OLAK instances), the reduced model reports a smaller optimality gap. However, the naïve model reports a smaller gap for BK and FL from OLAK instances. In comparison with the naïve model, we observe that the number of variables in the reduced model is decreased by at least 69.31% and 47.69% for the RCM instances and the OLAK instances, respectively. Interestingly, one can see that the number of variables is decreased by 99.74% for FL from RCM instances!

Figure 10. k-Core Decomposition



Notes. (Left) A social network of researchers; (right) the k-core decomposition of the network: white vertices have a coreness of two and gray vertices have a coreness of three. Therefore, $k_{med} = 2$ and $k_{max} = 3$.

Table 2. Results for Reduced Model vs. Naïve Model Under a 3,600-Second Time Limit (TL)

			Naïve model				Reduced model				
Abbreviation	k	b	#vars	B&B	Time (s)	Gap (%)	#vars	B&B	Time (s)	Gap (%)	Reduction in #vars (%)
FC	17	250	8,078	1	7.37	0.00	2,479	1	**0.86**	0.00	69.31
HP	4	250	24,012	1	5.45	0.00	6,049	1	**0.82**	0.00	74.81
FS	46	250	27,306	1,333	TL	1.69	8,085	11,269	TL	**1.22**	70.39
FN	33	250	27,764	7,852	TL	0.47	8,120	35,586	TL	**0.45**	70.75
CM	4	250	46,266	1	8.21	0.00	10,848	1	**1.25**	0.00	76.55
BK	2	250	116,456	1	16.40	0.00	25,887	29	**2.53**	0.00	77.77
FL	4	250	211,876	1	50.94	0.00	554	0	**33.77**	0.00	99.74
CA	21	250	299,368	2,255	TL	0.07	72,742	1	**1,083.53**	0.00	75.70
GW	3	250	393,182	1	108.17	0.00	100,437	1	**17.74**	0.00	74.46
CS	3	250	454,640	1	137.67	0.00	82,847	1,260	**18.42**	0.00	81.78
DB	3	250	634,160	1	203.94	0.00	124,059	1,950	**21.33**	0.00	80.44
DO	12	250	853,632	MEM	MEM	MEM	221,076	107	**487.00**	0.00	74.10
TH	17	250	913,262	MEM	MEM	MEM	234,446	3,879	**1,259.82**	0.00	74.33
GO	4	250	1,751,426	MEM	MEM	MEM	429,949	2,474	**288.57**	0.00	75.45
HU	5	250	3,949,310	MEM	MEM	MEM	959,493	1,515	**748.11**	0.00	75.70
BB	3	250	4,280,396	MEM	MEM	MEM	895,053	1	**967.96**	0.00	79.10
FC	20	20	8,078	33,101	396.51	0.00	2,672	62,619	**299.51**	0.00	66.92
BK	20	20	116,456	1,835	TL	**5.43**	60,920	2,171	TL	24.13	47.69
FL	20	20	211,876	1	2,913.83	**0.00**	95,668	69	TL	0.03	54.85
GW	20	20	393,182	30	TL	5.43	200,539	2,171	TL	**4.10**	49.00
DB	20	20	634,160	MEM	MEM	MEM	329,837	1	TL	**7.85**	47.99
YT	20	20	2,269,780	1	TL	3.53	1,138,333	14	TL	**1.35**	49.85

Notes. We report the number of branch-and-bound nodes (B&B), the percentage of the optimality gap (gap (%)), and the run time in seconds (time) for both models. The last column shows the percentage of reduction in number of variables. MEM denotes a memory crash during the IP solve process. Bold numbers represent best computational performances.

7.2. Reduced Model vs. Heuristic Approaches

Table 3 compares the computational performance of our reduced model against two existing heuristic approaches: OLAK (Zhang et al. 2017) and RCM (Laishram et al. 2020). We bold the best objective value and fastest time for each instance. For every instance, the reduced model has a strictly better (larger) objective value than OLAK and RCM. The reduced model is the superior approach even when the solver cannot prove optimality for FS and FN instances from the RCM instances (Laishram et al. 2020) and BK, FL, GW, DB, and YT instances from the OLAK instances (Zhang et al. 2017).

There are cases in which OLAK and RCM are terminated significantly faster than the reduced model, notably FS and FN from the RCM instances (Laishram et al. 2020) and FC from the OLAK instances (Zhang et al. 2017). However, we observe a considerable difference between the objective values of the reduced model and the heuristic approaches in the aforementioned instances. In a time limit of 3,600 seconds for the IP solver, we see the superiority of the reduced model over the heuristic approaches in (i) objective values for all instances; and (ii) *both* objective value and run time for 13 of 22 instances.

7.3. Experiments with Inequalities of Proposition 8

In this section, we test the practicality of the inequalities proposed in Proposition 8. Table 4 compares the performance of the reduced model without and with the inequalities. In our experiments, all of these inequalities are added upfront. Although we cannot conclude that the inequalities are helpful for RCM instances (above the horizontal line), we observe gap improvements for OLAK instances (below the horizontal line) when time-limit is reached (i.e., BK, FL, GW, DB, and YT). Furthermore, we see that the root LP relaxations are improved for most of the instances after adding these inequalities. We also observe a drastic decrease in number of the branch-and-bound nodes for CS, DB, DO and HU in the set of RCM instances.

7.4. Experiments with Supervalid Inequalities of Proposition 9

Table 5 summarizes the computational efficiency of the inequalities introduced in Proposition 9. In our computational experiments, all of these inequalities are added upfront. Although we observe gap improvements for FS and FN from the RCM instances and BK and GW from the OLAK instances, we see no remarkable time or gap improvement for other instances. Furthermore, we do not observe a significant improvement in the root LP relaxations. However, we see that adding the inequalities significantly decreases the number of branch-and-bound nodes for CS, DB, DO, and

Table 3. Results for Reduced Model vs. OLAK and RCM Heuristics

			Reduced model		OLAK		RCM	
Abbreviation	k	b	Objective	Time (s)	Objective	Time (s)	Objective	Time (s)
FC	17	250	**2,533**	0.86	2,225	117.90	2,472	**0.21**
HP	4	250	**6,978**	0.82	6,591	188.76	6,966	**0.66**
FS	46	250	**7,641**	TL	7,149	975.03	7,006	**16.92**
FN	33	250	**7,586**	TL	7,175	681.55	7,052	**12.28**
CM	4	250	**13,939**	**1.25**	13,802	188.76	13,911	1.96
BK	2	250	**35,267**	**2.53**	35,140	478.00	35,266	5.82
FL	4	250	**105,388**	33.77	105,391	5,591.92	105,293	**20.53**
CA	21	250	**79,561**	1,083.53	79,201	9,656.34	79,395	**1,071.30**
GW	3	250	**107,299**	**17.74**	107,082	1,984.72	107,260	114.44
CS	3	250	**153,765**	**18.42**	153,477	2,135.08	153,677	97.49
DB	3	250	**204,857**	**21.33**	204,529	2,865.69	204,775	169.02
DO	12	250	**215,343**	**487.00**	214,806	16,824.92	215,197	551.24
TH	17	250	**232,329**	1,259.82	231,569	26,830.27	232,025	**1,249.98**
GO	4	250	**494,579**	**288.57**	493,229	11,167.51	494,044	5,015.32
HU	5	250	**1,056,519**	**748.11**	1,055,814	36,594.14	1,056,330	3,827.08
BB	3	250	**1,278,551**	**967.96**	1,278,231	48,052.84	1,278,526	2,327.67
FC	20	20	**1,967**	299.51	1,894	9.76	1,902	**1.81**
BK	20	20	**1,181**	TL	998	20.46	957	**1,601.90**
FL	20	20	**15,833**	TL	15,822	405.87	MEM	MEM
GW	20	20	**8,433**	TL	8,161	103.62	MEM	MEM
DB	20	20	**3,123**	TL	3,066	116.67	MEM	MEM
YT	20	20	**19,088**	TL	18,939	320.71	MEM	MEM

Notes. We report the best objective value (obj) and the run time in seconds (time). We set a time limit of 3,600 seconds for the reduced IP model. MEM denotes a memory crash during the heuristic process. Bold numbers represent best computational performances.

Table 4. Results for the Reduced Model (2) Without and with Inequalities of Proposition 8 Under a 3,600-Second Time Limit (TL)

			Reduced model w/o inequalities				Reduced model w/ inequalities					
Abbreviation	k	b	Root	B&B	Time (s)	Gap (%)	#ineq	Root	B&B	Time (s)	Gap (%)	Root
FC	17	250	2,551.96	1	0.86	0.00	683	2,549.43	1	0.01	0.75	0.00
HP	4	250	7,114.50	1	0.82	0.00	748	7,065.88	1	0.02	0.83	0.00
FS	46	250	8,317.44	11,269	TL	1.22	1,396	8,264.02	8,617	0.02	TL	1.10
FN	33	250	8,187.01	35,586	TL	0.45	1,136	8,125.08	36,452	0.02	TL	0.42
CM	4	250	14,343.84	1	1.25	0.00	1,400	14,141.32	1	0.03	1.27	0.00
BK	2	250	36,099.50	29	2.53	0.00	2,360	35,400.50	1	0.07	2.43	0.00
FL	4	250	105,388.00	0	33.77	0.00	3	105,388.00	0	0.12	33.28	0.00
CA	21	250	80,801.72	1	1,083.53	0.00	1,894	80,287.70	1	0.16	1,097.95	0.00
GW	3	250	113,017.29	1	17.7	0.00	11,230	108,957.15	1	0.28	18.88	0.00
CS	3	250	157,856.31	1,260	18.42	0.00	8,966	155,390.27	1	0.36	17.49	0.00
DB	3	250	210,415.12	1,950	21.33	0.00	11,353	206,937.42	1	0.47	19.44	0.00
DO	12	250	222,144.93	107	487.00	0.00	9,207	218,699.11	1	0.49	489.30	0.00
TH	17	250	239,766.52	3,879	1,259.82	0.00	9,664	236,349.85	2,604	0.51	1,199.50	0.00
GO	4	250	519,086.05	2,474	288.57	0.00	45,995	508,224.64	2,020	1.30	308.73	0.00
HU	5	250	1,086,242.55	1,515	748.11	0.00	42,459	1,064,290.07	1	2.44	759.97	0.00
BB	3	250	1,297,872.26	1	967.96	0.00	34,924	1,281,799.13	1	2.58	988.03	0.00
FC	20	20	2,275.17	62,619	299.51	0.00	745	2,228.08	55,105	0.01	273.12	0.00
BK	20	20	3,524.91	2,171	TL	24.13	4,005	3,368.78	4,055	0.05	TL	19.45
FL	20	20	16,305.32	69	TL	0.03	8,529	16,237.78	54	0.11	TL	0.01
GW	20	20	16,391.29	2,171	TL	4.10	16,694	15,663.16	1,327	0.18	TL	3.80
DB	20	20	7,285.80	1	TL	7.85	26,347	6,970.54	1	0.28	TL	5.71
YT	20	20	32,862.46	14	TL	1.35	25,026	31,579.82	15	0.40	TL	1.31

Notes. We report the root LP relaxation (root), number of variables (#vars), number of branch-and-bound nodes (B&B), preprocess time to find inequalities in seconds (ptime), time to solve the IP model in seconds (IP time), and the percentage of the optimality gap (gap (%)) for both models. The number of the added inequalities is shown by #ineq.

Table 5. Experiments with the Supervalid Inequalities of Proposition 9 Under IP Time Limit of 3,600 Seconds (TL)

			Reduced model w/o inequalities				Reduced model w/ inequalities					
Abbreviation	k	b	Root	B&B	IP time (s)	Gap (%)	#ineq	Root	B&B	IP time (s)	Gap (%)	Root
FC	17	250	2,551.96	1	0.86	0.00	3,049	2,551.96	1	0.86	1.00	0.00
HP	4	250	7,114.50	1	0.82	0.00	2,062	7,114.50	1	0.12	0.88	0.00
FS	46	250	8,317.44	11,269	TL	1.22	8,188	8,317.44	9,111	26.16	TL	1.07
FN	33	250	8,187.01	35,586	TL	0.45	7,184	8,187.01	28,929	5.94	TL	0.39
CM	4	250	14,343.84	1	1.25	0.00	4,339	14,343.73	1	0.23	1.59	0.00
BK	2	250	36,099.50	29	2.53	0.00	1,587	36,099.50	33	0.20	2.64	0.00
FL	4	250	105,388.00	0	33.77	0.00	4	105,388.00	0	0.01	34.55	0.00
CA	21	250	80,801.72	1	1,083.53	0.00	18,587	80,801.72	12	2.36	1,112.24	0.00
GW	3	250	113,017.29	1	17.74	0.00	17,892	113,014.90	1	1.12	18.56	0.00
CS	3	250	157,856.31	1,260	18.42	0.00	22,010	157,856.05	1	1.3	17.6	0.00
DB	3	250	210,415.12	1,950	21.33	0.00	25,785	210,412.36	1	1.71	22.04	0.00
DO	12	250	222,144.94	107	487.00	0.00	56,461	222,144.91	1	4.93	494.11	0.00
TH	17	250	239,766.52	3,879	1,259.82	0.00	101,693	239,766.50	2,269	14.48	1,378.69	0.00
GO	4	250	519,086.05	2,474	288.57	0.00	512,588	519,085.75	1,721	37.76	979.75	0.00
HU	5	250	1,086,242.55	1,515	748.11	0.00	108,707	1,086,242.44	1	31.18	820.09	0.00
BB	3	250	1,297,872.26	1	967.96	0.00	40,722	1,297,871.65	1	9.00	1,047.43	0.00
FC	20	20	2,275.17	62,619	299.51	0.00	3,662	2,275.17	98,168	1.28	564.05	0.00
BK	20	20	3,524.91	2,171	TL	24.13	325,652	3,524.91	1,144	87.07	TL	18.51
FL	20	20	16,305.32	69	TL	0.03	413,293	16,305.32	1	2,717.32	2,620.42	0.00
GW	20	20	16,391.29	2,171	TL	4.10	559,498	16,391.29	30	132.68	TL	3.94
DB	20	20	7,285.80	1	TL	7.85	1,332,530	7,285.80	1	469.37	TL	83.75
YT	20	20	32,862.46	14	TL	1.35	12,136,498	24,721.94	1	861.10	TL	23.50

Notes. We report the LP root relaxation (root), number of branch-and-bound nodes (B&B), preprocess time to find inequalities in seconds (ptime), time to solve the IP model in seconds (IP time), and the percentage of the optimality gap (gap (%)). Column #ineq shows the number of supervalid inequalities added to the reduced model.

Table 6. Experiments with the Fixing Procedure of Proposition 10 Under IP Time Limit of 3,600 Seconds (TL)

			Reduced model w/o fixing				Reduced model w/ fixing					Reduction in
Abbreviation	k	b	#vars	B&B	IP time (s)	Gap (%)	#vars	B&B	ptime	IP time (s)	Gap (%)	#vars (%)
FC	17	250	2,479	1	0.86	0.00	2,479	1	0.08	0.87	0.00	0.00
HP	4	250	6,049	1	0.82	0.00	5,301	1	0.12	0.83	0.00	12.37
FS	46	250	8,085	11,269	TL	1.22	7,880	8,699	0.56	TL	1.07	2.54
FN	33	250	8,120	35,586	TL	0.45	7,881	37,310	0.39	TL	0.43	2.94
CM	4	250	10,848	1	1.25	0.00	9,243	1	0.12	1.23	0.00	14.80
BK	2	250	25,887	29	2.53	0.00	25,057	29	0.27	2.53	0.00	3.21
FL	4	250	554	0	33.77	0.00	376	0	3.18	34.04	0.00	32.13
CA	21	250	72,742	1	1,083.53	0.00	69,460	1	5.65	1,062.47	0.00	4.51
GW	3	250	100,437	1	17.74	0.00	96,446	1	1.26	17.78	0.00	3.97
CS	3	250	82,847	1,260	18.42	0.00	81,277	1,716	1.23	19.95	0.00	1.90
DB	3	250	124,059	1,950	21.33	0.00	122,139	1,950	1.63	20.86	0.00	1.55
DO	12	250	221,076	107	487.00	0.00	213,831	107	10.49	476.43	0.00	3.28
TH	17	250	234,446	3,879	1,259.82	0.00	230,594	3,879	18.1	1,262.45	0.00	1.64
GO	4	250	429,949	2,474	288.57	0.00	410,349	2,747	6.35	295.82	0.00	4.56
HU	5	250	959,493	1,515	748.11	0.00	926,618	1,515	24.80	768.75	0.00	3.43
BB	3	250	895,053	1	967.96	0.00	847,462	1	30.57	995.88	0.00	5.32
FC	20	20	2,672	62,619	299.51	0.00	2,671	62,619	0.07	310.30	0.00	0.04
BK	20	20	60,920	2,171	TL	24.13	40,544	3,352	0.21	TL	19.15	33.45
FL	20	20	95,668	69	TL	0.03	88,994	69	2.88	TL	0.03	6.98
GW	20	20	200,539	2,171	TL	4.10	139,488	2,161	0.92	TL	3.94	30.44
DB	20	20	329,827	1	TL	7.85	215,787	1	0.98	TL	7.27	34.58
YT	20	20	1,138,333	14	TL	1.35	743,326	14	3.75	TL	1.33	34.70

Notes. We report number of variables (#vars), number of branch-and-bound nodes (B&B), preprocess time to find inequalities in seconds (ptime), time to solve the IP model in seconds (IP time), and the percentage of the optimality gap (gap (%)). Last column shows the percentage of reduction in number of variables.

Table 7. Experiments with Fixing Procedure in Proposition 11 for $k = 20$ and $b \in \{1,5,10,15\}$ Under IP Time Limit of 3,600 Seconds (TL)

			Reduced model w/o fixing				Reduced model w/ fixing					Reduction in
Abbreviation	k	b	#vars	B&B	Time (s)	Gap (%)	#vars	B&B	ptime	IP time (s)	Gap (%)	#vars (%)
FC	20	1	2,672	1	0.44	0.00	2,241	1	0.40	1.77	0.00	16.13
		5		3,514	7.73	0.00	2,516	2,576	0.38	2.83	0.00	5.84
		10		10,320	30.13	0.00	2,647	6,472	0.17	27.12	0.00	0.94
		15		28,750	83.31	0.00	2,661	31,880	0.17	64.83	0.00	0.41
BK	20	1	60,920	1	7.14	0.00	57,498	1	0.76	2.97	0.00	5.62
		5		19,304	TL	10.84	58,690	27,546	0.74	TL	11.70	3.66
		10		4,670	TL	15.13	60,189	2,614	0.74	TL	14.79	1.20
		15		5,376	TL	17.37	60,833	3,262	0.24	TL	18.95	0.14
GW	20	1	200,539	1	44.16	0.00	189,269	1	5.61	10.2	0.00	5.62
		5		13,199	TL	1.17	193,384	29,673	5.37	TL	1.42	3.57
		10		2,853	TL	3.43	197,822	2,559	5.34	TL	2.26	1.35
		15		2,886	TL	3.62	200,138	2,232	2.22	TL	3.68	0.20
DB	20	1	329,837	1	188.82	0.00	314,034	0	2.13	8.57	0.00	4.79
		5		699	TL	2.31	314,284	4,451	2.90	10.12	0.00	4.72
		10		1	TL	3.75	316,945	1,985	4.51	TL	1.89	3.91
		15		1	TL	5.52	326,178	25	2.96	TL	5.15	1.11
YT	20	1	1,138,333	1	177.97	0.00	1,117,600	1	14.24	47.31	0.00	1.82
		5		1,580	TL	0.48	1,124,150	10,311	13.91	TL	0.20	1.25
		10		34	TL	1.09	1,131,648	1,693	12.90	TL	1.05	0.59
		15		21	TL	1.18	1,136,342	26	9.07	TL	1.18	0.17

Notes. We report number of variables (#vars), number of branch-and-bound nodes (B&B), preprocess time to fix x variables in seconds (ptime), time to solve the IP model in seconds (IP time), and the percentage of the optimality gap (gap (%)) for both models.

HU from the RCM instances. Interestingly, one can notice that FL from the OLAK instances can be solved to optimality at the root node of the branch-and-bound tree after 2,620.42 seconds; however, this requires 2,717.32 seconds of preprocess for adding the inequalities.

Table 8. Experiments with Best Computational Improvements Under IP Time Limit of 3,600 Seconds (TL)

			Reduced model w/o improvements				Reduced model w/ improvements				
Abbreviation	k	b	Root	B&B	IP time (s)	Gap (%)	Root	B&B	ptime	IP time (s)	Gap (%)
FC	17	250	2,551.96	1	0.86	0.00	2,549.43	1	0.08	0.77	0.00
HP	4	250	7,114.50	1	0.82	0.00	7,065.88	1	0.14	0.81	0.00
FS	46	250	8,317.44	11,269	TL	1.22	8,264.02	9,180	0.57	TL	1.07
FN	33	250	8,187.01	35,586	TL	0.45	8,125.08	30,346	0.42	TL	0.39
CM	4	250	14,343.84	1	1.25	0.00	14,141.32	1	0.14	1.23	0.00
BK	2	250	36,099.50	29	2.53	0.00	35,400.50	1	0.35	2.39	0.00
FL	4	250	105,388.00	0	33.77	0.00	105,388	0	3.33	32.81	0.00
CA	21	250	80,801.72	1	1,083.53	0.00	80,287.70	1	5.88	1,067.39	0.00
GW	3	250	113,017.30	1	17.74	0.00	108,957.15	1	1.57	18.53	0.00
CS	3	250	157,856.31	1,260	18.42	0.00	155,390.27	1	1.61	13.68	0.00
DB	3	250	210,415.12	1,950	21.33	0.00	206,937.42	1	2.09	19.22	0.00
DO	12	250	222,144.93	107	487.00	0.00	218,799.11	1	10.85	471.50	0.00
TH	17	250	239,766.52	3,879	1,259.82	0.00	236,349.85	2,604	17.95	1,211.39	0.00
GO	4	250	519,086.05	2,474	288.57	0.00	508,224.64	2,020	7.30	318.20	0.00
HU	5	250	1,086,242.55	1,515	748.11	0.00	1,064,290.07	1	26.43	753.17	0.00
BB	3	250	1,297,872.26	1	967.96	0.00	1,281,799.13	1	32.70	994.65	0.00
FC	20	20	2,275.17	62,619	299.51	0.00	2,228.08	55,105	0.08	271.84	0.00
BK	20	20	3,524.91	2,171	TL	24.13	3,368.78	4,057	0.25	TL	19.45
FL	20	20	16,305.32	69	TL	0.03	16,237.78	54	2.92	TL	0.01
GW	20	20	16,391.29	2,171	TL	4.10	15,663.16	1,308	1.10	TL	3.80
DB	20	20	7,285.80	1	TL	7.85	6,970.54	1	1.27	TL	5.71
YT	20	20	32,862.46	14	TL	1.35	31,579.82	15	4.16	TL	1.31

Notes. We report number of variables (#vars), number of branch-and-bound nodes (B&B), preprocess time to find inequalities in seconds (ptime), time to solve the IP model in seconds (IP time), and the percentage of the optimality gap (gap (%)). Last column shows the percentage of reduction in number of variables.

7.5. Experiments with Fixing Procedure of Proposition 10

Table 6 reports the fixing percentages and computational performance of the reduced model with the fixing procedure of Proposition 10. We observe that the fixing procedure fixes at most 32.13% and 34.70% of the y variables of the reduced model for the RCM and the OLAK instances, respectively. When time limit is reached, we see that the fixing procedure reduces the optimality gap for FS and FN from the RCM instances and BK, GW, DB, and YT from the OLAK instances. Nevertheless, we do not observe a significant change in run times when the problem is solved in time limit.

7.6. Experiments with Fixing Procedure in Proposition 11

In this section, we test the computational performance of the fixing procedure presented in Proposition 11. To respect the condition of the proposition, we consider instances with $k = 5$ and $b \in \{1, 5, 10, 15\}$ that are also reported by Zhang et al. (2017). Although the percentage of fixing is not significant for DB and YT, we observe that the fixing procedure helps decrease either the solve time or the optimality gap. Interestingly, the fixing procedure makes DB with $k = 20$ and $b = 5$ solvable in just 13.02 seconds.

7.7. Experiments with Best Computational Improvements

Based on our computational experiments with inequalities and fixing procedures proposed in Section 6, we conduct a final set of experiments with "best" of them: (i) inequalities of Proposition 8 and (ii) fixing procedure of Proposition 10. Table 8 summarizes the computational performance of the reduced model with the aforementioned inequalities and fixing procedure. For each instance, a combination of these procedures does not work better than the best of them.

8. Conclusion and Future Work

In this paper, we propose an integer programming model for solving the anchored k-core problem that is known as a hard combinatorial optimization problem. The number of decision variables in the new IP formulation is at least half of the number of decision variable in a naïve model of the problem. Because of the small size of the proposed IP formulation, we prove that the convex hull of all the feasible points of the problem form a full-dimensional polytope in the reduced space. Furthermore, we show that (i) the LP relaxation of the proposed model is at least as strong as that of the naïve formulation, and (ii) multiple inequalities of the reduced IP model are facet-defining under reasonable and mild conditions. Our numerical results show the computational superiority of our proposed IP formulation over the naïve one and two existing heuristics in the literature. To improve the computational performance of the reduced IP model, we develop further valid and supervalid inequalities as well as fixing procedures.

For future work, one can focus on developing novel integer programming techniques (e.g., decomposition methods and new valid inequalities and fixings) to solve the unsolved instances to optimality. Another direction can be studying other variants of the anchored k-core problem (e.g., edge addition, edge deletion, and vertex deletion) from the lens of operations research.

Acknowledgments

The authors thank Logan Smith for helpful early discussions and the anonymous reviewers for helpful comments.

Endnotes

[1] For clarity and consistency purposes, we prefer to use "engagement" terminology rather than "resilience" throughout the paper.

[2] See https://meetings.informs.org/wordpress/indianapolis2022/.

[3] In many real-world benchmark instances of the maximum anchored k-core problem, we observe that $b \geq k$ holds. For example, see instances of Zhang et al. (2017) and Laishram et al. (2020).

[4] See https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.core.k_core.html.

References

Adger WN (2000) Social and ecological resilience: Are they related? *Progress Human Geography* 24(3):347–364.

Al-garadi MA, Varathan KD, Ravana SD (2017) Identification of influential spreaders in online social networks using interaction weighted k-core decomposition method. *Phys. A* 468:278–288.

Altaf-Ul-Amine M, Nishikata K, Korna T, Miyasato T, Shinbo Y, Arifuzzaman M, Wada C, et al. (2003) Prediction of protein functions based on k-cores of protein-protein interaction networks and amino acid sequences. *Genome Informatics* 14:498–499.

Bader GD, Hogue CW (2002) Analyzing yeast protein–protein interaction data obtained from different sources. *Nature Biotech.* 20(10):991–997.

Balasundaram B, Butenko S, Hicks IV (2011) Clique relaxations in social network analysis: The maximum k-plex problem. *Oper. Res.* 59(1):133–142.

Batagelj V, Zaversnik M (2003) An $O(m)$ algorithm for cores decomposition of networks. *CoRR* cs.DS/0310049.

Bhawalkar K, Kleinberg J, Lewi K, Roughgarden T, Sharma A (2015) Preventing unraveling in social networks: The anchored k-core problem. *SIAM J. Discrete Math.* 29(3):1452–1475.

Bonchi F, Gullo F, Kaltenbrunner A, Volkovich Y (2014) Core decomposition of uncertain graphs. *Proc. 20th ACM SIGKDD Internat. Conf. Knowledge Discovery Data Mining* (ACM, New York), 1316–1325.

Burleson-Lesser K, Morone F, Tomassone MS, Makse HA (2020) *k*-core robustness in ecological and financial networks. *Sci. Rep.* 10(1):1–14.

Christof T, Loebel A (2022) Polyhedron representation transformation algorithm. Accessed July 17, 2022, http://porta.zib.de.

Conforti M, Cornuéjols G, Zambelli G (2014) *Integer Programming*, vol. 271 (Springer, Berlin).

Daemi N, Borrero JS, Balasundaram B (2022) Interdicting low-diameter cohesive subgroups in large-scale social networks. *INFORMS J. Optim.* 4(3):304–325.

Daianu M, Jahanshad N, Nir TM, Toga AW, Jack CR, Weiner MW, Thompson PM, for the Alzheimer's Disease Neuroimaging Initiative (2013) Breakdown of brain connectivity between normal aging and Alzheimer's disease: A structural *k*-core network analysis. *Brain Connectivity* 3(4):407–422.

Dey P, Maity SK, Medya S, Silva A (2020) Network robustness via global *k*-cores. Preprint, submitted December 18, https://arxiv.org/abs/2012.10036.

Gao J, Xu Z, Li R, Yin M (2022) An exact algorithm with new upper bounds for the maximum *k*-defective clique problem in massive sparse graphs. *Proc. AAAI Conf. Artificial Intelligence* 36(9):10174–10183.

Garas A, Schweitzer F, Havlin S (2012) A *k*-shell decomposition method for weighted networks. *New J. Phys.* 14(8):083030.

Garcia D, Mavrodiev P, Schweitzer F (2013) Social resilience in online communities: The autopsy of friendster. *Proc. 1st ACM Conf. Online Social Networks* (Association for Computing Machinery, New York), 39–50.

Giatsidis C, Thilikos DM, Vazirgiannis M (2011a) Evaluating cooperation in communities with the *k*-core structure. *Proc. Internat. Conf. Adv. Social Networks Analysis Mining* (IEEE, Piscataway, NJ), 87–93.

Giatsidis C, Thilikos DM, Vazirgiannis M, Giatsidis C, Vazirgiannis M, Thilikos DM (2011b) D-cores: Measuring collaboration of directed graphs based on degeneracy. *Proc. IEEE Internat. Conf. Data Mining* (IEEE, Piscataway, NJ).

Hagmann P, Cammoun L, Gigandet X, Meuli R, Honey CJ, Wedeen VJ, Sporns O (2008) Mapping the structural core of human cerebral cortex. *PLoS Biology* 6(7):e159.

Israeli E, Wood RK (2002) Shortest-path network interdiction. *Networks* 40(2):97–111.

Laishram R, Erdem Sar A, Eliassi-Rad T, Pinar A, Soundarajan S (2020) Residual core maximization: An efficient algorithm for maximizing the size of the *k*-core. *Proc. SIAM Internat. Conf. Data Mining* (SIAM, Philadelphia), 325–333.

Leskovec J, Krevl A (2014) SNAP datasets: Stanford large network data set collection. Accessed July 17, 2022, http://snap.stanford.edu/data.

Ma J, Balasundaram B (2019) On the chance-constrained minimum spanning *k*-core problem. *J. Global Optim.* 74(4):783–801.

Ma J, Mahdavi Pajouh F, Balasundaram B, Boginski V (2016) The minimum spanning *k*-core problem with bounded CVaR under probabilistic edge failures. *INFORMS J. Comput.* 28:295–307.

Malliaros FD, Vazirgiannis M (2013) To stay or not to stay: Modeling engagement dynamics in social graphs. *Proc. 22nd ACM Internat. Conf. Inform. Knowledge Management* (ACM, New York), 469–478.

Matula D, Beck L (1983) Smallest-last ordering and clustering and graph coloring algorithms. *J. ACM* 30(3):417–427.

Miao Z, Balasundaram B (2020) An ellipsoidal bounding scheme for the quasi-clique number of a graph. *INFORMS J. Comput.* 32(3):763–778.

Mikesell D, Hicks IV (2022) Minimum *k*-cores and the *k*-core polytope. *Networks* 80(1):93–108.

Nemhauser GL, Trotter LE (1974) Properties of vertex packing and independence system polyhedra. *Math. Programming* 6(1):48–61.

Pajouh FM, Balasundaram B, Hicks IV (2016) On the 2-club polytope of graphs. *Oper. Res.* 64(6):1466–1481.

Pattillo J, Youssef N, Butenko S (2013) On clique relaxation models in network analysis. *Eur. J. Oper. Res.* 226(1):9–18.

Pei S, Muchnik L, Andrade J Jr, Zheng Z, Makse HA (2014) Searching for superspreaders of information in real-world social media. *Sci. Rep.* 4(1):5547.

Peng Y, Zhang Y, Zhang W, Lin X, Qin L (2018) Efficient probabilistic *k*-core computation on uncertain graphs. *Proc. IEEE 34th Internat. Conf. Data Engrg.* (IEEE, Piscataway, NJ), 1192–1203.

Qin L, Wang Y, Sun Q, Zhang X, Shia BC, Liu C (2020) Analysis of the COVID-19 epidemic transmission network in mainland China: *k*-core decomposition study. *JMIR Public Health Surveillance* 6(4):e24291.

Rivlin G (2006) Wallflower at the web party. *New York Times* (October 15), https://www.nytimes.com/2006/10/15/business/yourmoney/wallflower-at-the-web-party.html.

Rossi R, Ahmed N (2015) The network data repository with interactive graph analytics and visualization. *Proc. AAAI Conf. Artificial Intelligence* 29(1).

Salemi H, Buchanan A (2020) Parsimonious formulations for low-diameter clusters. *Math. Programming Comput.* 12(3):493–528.

Seidman SB (1983) Network structure and minimum degree. *Soc. Networks* 5(3):269–287.

Seki K, Nakamura M (2016) The collapse of the friendster network started from the center of the core. *Proc. IEEE/ACM Internat. Conf. Adv. Social Networks Analysis Mining* (IEEE Piscataway, NJ), 477–484.

Shanahan M, Bingman VP, Shimizu T, Wild M, Güntürkün O (2013) Large-scale network organization in the avian forebrain: A connectivity matrix and theoretical analysis. *Frontiers Comput. Neurosci.* 7:89.

Tootoonchi B, Srinivasan V, Thomo A (2017) Efficient implementation of anchored 2-core algorithm. *Proc. 2017 IEEE/ACM Internat. Conf. Adv. Social Networks Analysis Mining* (Association for Computing Machinery, New York), 1009–1016.

Verma A, Buchanan A, Butenko S (2015) Solving the maximum clique and vertex coloring problems on very large sparse networks. *INFORMS J. Comput.* 27(1):164–177.

Wood CI, Hicks IV (2015) The minimal *k*-core problem for modeling *k*-assemblies. *J. Math. Neurosci.* 5(1):1–19.

Wu S, Das Sarma A, Fabrikant A, Lattanzi S, Tomkins A (2013) Arrival and departure dynamics in social networks. *Proc. 6th ACM Internat. Conf. Web Search Data Mining* (Association for Computing Machinery, New York), 233–242.

Wu H, Cheng J, Lu Y, Ke Y, Huang Y, Yan D, Wu H (2015) Core decomposition in large temporal graphs. *Proc. IEEE Internat. Conf. Big Data* (IEEE, Piscataway, NJ), 649–658.

Zhang F, Zhang W, Zhang Y, Qin L, Lin X (2017) OLAK: An efficient algorithm to prevent unraveling in social networks. *Proc. VLDB Endowment* 10(6):649–660.

Zhou Z, Zhang F, Lin X, Zhang W, Chen C (2019) *k*-core maximization: An edge addition approach. *Proc. Twenty-Eighth Internat. Joint Conf. Artificial Intelligence (IJCAI-19)* (International Joint Conferences on Artificial Intelligence Organization), 4867–4873.