IISE Transactions

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/uiie21

The influence coverage optimization problem

Majid Akhgar & Juan S. Borrero

To cite this article: Majid Akhgar & Juan S. Borrero (2024) The influence coverage optimization problem, IISE Transactions, 56:11, 1162-1175, DOI: 10.1080/24725854.2023.2261507

To link to this article: https://doi.org/10.1080/24725854.2023.2261507

IISE TRANSACTIONS
2024, VOL. 56, NO. 11, 1162–1175
https://doi.org/10.1080/24725854.2023.2261507

Taylor & Francis
Taylor & Francis Group

Check for updates

The influence coverage optimization problem

Majid Akhgar and Juan S. Borrero

School of Industrial Engineering and Management, Oklahoma State University, Stillwater, OK, USA

ABSTRACT
We introduce the Influence Coverage Optimization Problem (ICOP), which is an influence maximization problem where the activation of nodes also depends on their location on the plane. Specifically, the ICOP assumes that there is a network where nodes become active (i.e., influenced) either by the influence they receive from interactions with active in-neighbors or by entering the coverage area of a physical ad or a Geo-fence. The objective is to locate a fixed number of ads or Geo-fences and modify the network influence rates to minimize the network activation time. Assuming a Markovian influence model, we prove that the ICOP is *NP*-hard, and then we present mixed-integer programming formulations for three different types of coverage modes. A reformulation of the non-linear "big-M" constraints, two types of valid cuts, and a fast heuristic based on the *k*-means algorithm are used as enhancements that facilitate solving the ICOP via an Iterative Decomposition Branch-and-Cut (IDBC) algorithm. In addition, we present an alternative discrete formulation of the ICOP using critical intersection points. Several experiments under various parameter configurations across instances with more than a hundred nodes and thousand arcs are conducted, showing the IDBC's capability to provide optimal solutions within seconds or minutes for most instances. Moreover, the experiments reveal that the ICOP can significantly outperform a Geo-fence coverage model that does not consider network interactions to make location decisions.

ARTICLE HISTORY
Received 20 April 2022
Accepted 30 August 2023

KEYWORDS
Influence maximization; social networks; maximum coverage; critical intersection points; branch-and-cut

1. Introduction

Models of cascading behaviors have gained widespread attention in recent years, due to the proliferation of physical networks and online communication in social networks (Borge-Holthoefer *et al.*, 2013). These models explore how ideas, information, rumors, and other phenomena spread through complex interconnected environments (Hethcote, 2000; Pastor-Satorras and Vespignani, 2001; Kempe *et al.*, 2003, 2015; Kitsk *et al.*, 2010; Nowzari *et al.*, 2016). Applications such as viral marketing have been particularly well-studied in this field (Domingos and Richardson, 2001; Domingos, 2005), as companies use cascading behavior to promote new products by targeting key customers who then spread the information throughout the network using the so-called "word-of-mouth" mechanism. This can create a "snowball" effect that eventually reaches the entire network.

In some cases, it is sufficient to consider only the network topology and the strength of connections when determining influence strategies. However, in many situations, determining effective strategies requires additional information, such as the location and movement of customers. As an example, a company looking to place physical ads in a city, for example billboards or flyers, should consider the geographical location of their potential customers to start a propagation process. For instance, it may be more beneficial to place the ads close to places that are frequently visited by *influential* trend-setting customers, rather than in areas visited by many non-influential customers.

Another example is a company utilizing *Geo-fences*, which are virtual regions in the shape of circles with a fixed coverage radius placed over a geographical area. When individuals enter or leave the fence, they receive messages such as ads, coupons, and recommendations on their smartphones (Rodriguez Garzon and Deva, 2014; Berman, 2016; Arief *et al.*, 2020; Ho *et al.*, 2020). Geo-fences can be used in location-based virtual marketing strategies and are a cost-effective alternative to physical ad-placement (Küpper *et al.*, 2011). As seen in Figure 1, a retailer can place a Geo-fence around its physical store or that of its competition, and customers receive discount coupons as they enter the Geo-fence. In this scenario, a company may be interested in finding the optimal location of a fixed number of Geo-fences in a geographical region to optimize the spread of information about its product among potential customers. As with physical ads, the placement decisions should be based on the geographic location of customers and the network relationships between them.

This article studies the Influence-Coverage Optimization Problem (ICOP), which combines network influence maximization with planar facility location problems (Church, 1984; Murray and Tong, 2007; Bansal and Kianfar, 2017).

CONTACT Juan S. Borrero juan.s.borrero@okstate.edu
Supplemental data for this article can be accessed online at https://doi.org/10.1080/24725854.2023.2261507.

Figure 1. Geo-fence concept for retailers.

The problem involves an underlying social network where nodes (i.e., customers) are activated (i.e., influenced) by their active in-neighbors or by entering the coverage area of a physical ad or a Geo-fence. Unlike standard facility location models, which only consider coverage of points, the ICOP also takes into account the typical movement of the node, for example, the sequence of movements between home, office, and cafe. In this context, a node is influenced by an ad or a Geo-fence if at least one point of the individual's trajectory is within the radius of coverage of the ad or Geo-fence. The goal of the ICOP is to locate a fixed number of ads or Geo-fences and invest resources encouraging "word-of-mouth" spread in order to minimize the time until all nodes in the network are active.

Similar to the models in Borrero *et al.* (2018) and Borrero *et al.* (2021), we assume that the influence times are stochastically distributed in the ICOP and use a Continuous-Time Markov Chain to track the activation status of the nodes. However, in contrast with previous models, the external influence of nodes in the ICOP depends on the location of the ads or Geo-fences in the area of interest. In the ICOP, we consider three different modes of coverage: point-wise or static coverage, segment coverage, and piece-wise linear coverage (detailed in Section 3.1); and use Mixed-Integer Programming (MIP) approach to formulate the resulting stochastic influence and location optimization problem.

Standard MIP solvers are unable to directly and reliably solve the MIP formulations that result from our problem, due to an exponential number of linear constraints and due to non-linear constraints with "big-M" constants. To address this issue, we propose formulation enhancements and a heuristic that quickly finds good feasible solutions. We use the enhancements and the heuristic to design an Iterative Decomposition Branch-and-Cut algorithm (IDBC) that deals with the exponentially many constraints "on the fly", and iteratively adds non-linear constraints as needed. Furthermore, we study an alternative formulation based on the concept of *critical intersection points* (Church, 1984). This formulation avoids the non-linear constraints and the "big-Ms" of the original formulation, with the price of increasing the number of binary variables.

We summarize our main contributions as follows:

1. Our proposed optimization model for network influence maximization is novel in the literature, as it considers the location of the nodes and the placement of ads or Geo-fences as influence triggers. To the best of our knowledge, this model is the first of its kind in the literature of influence maximization, as it makes decisions on both the locations and rates of receiving influence from in-neighbors simultaneously, with the goal of maximizing the spread of influence.
2. We prove that the resulting optimization problem is *NP*-hard and present an MIP approach formulation under each of three different coverage modes. To improve the formulations, we derive two sets of valid cuts that exploit the geometric properties of the problem and reformulate the non-linear "big-M" constraints as linear constraints. Furthermore, we derive a heuristic based on the *k*-means algorithm (Likas *et al.*, 2003) to quickly find feasible solutions.
3. We provide an exact optimization algorithm to solve the problem that iteratively solves a large-scale MIP using a Decomposition Branch-and-Cut (DBC) algorithm. This approach is able to provide optimal and numerically stable solutions for instances up to hundreds of nodes in a short amount of time, usually within seconds to minutes, depending on the parameter configurations used. As shown in our numerical experiments, the IDBC solution times are orders of magnitude faster than those of other cutting plane approaches. Particularly, the IDBC solves most of our test instances to optimality, whereas a basic cutting plane algorithm with no enhancements is unable to find feasible solutions within the time limit. In addition, the IDBC significantly outperforms the discrete formulation based on critical intersection points, especially in larger instances.

The rest of this article is organized as follows: Section 2 has the literature review, Sections 3 and 4 discuss the ICOP formulations and enhancements. The IDBC is in Sections 5 and the discrete formulation in Section 6. The computational experiments are in Section 7. The proofs are provided in Appendix II.

2. Literature review

2.1. General influence maximization models

Kempe *et al.* (2003) introduced two models for Influence Maximization (IM) in networks, referred to as the threshold model and the independent cascade model. The goal of these models is to select key nodes to influence, known as the seeds of propagation, in order to achieve the maximum expected number of active nodes at the end of the cascade. The authors demonstrated that such problems are *NP*-hard

and proposed approximation algorithms based on the submodularity of the objective function. Evaluating the submodular function can be challenging, leading to extensive research on these models, as seen in studies such as Chen *et al.* (2013) and Kempe *et al.* (2015).

Wu and Küçükyavuz (2018) and Güney (2019) proposed two-stage stochastic programs and an integer programming formulation, respectively, for the IM problem. Güney *et al.* (2021) further reformulated the problem by utilizing the stochastic maximal covering location problem and applied a Benders decomposition algorithm to solve the problem. Additionally, researchers investigated two-player extensions of the IM problem and employed techniques such as heuristics, stochastic optimization, and bilevel optimization to solve these problems, see Tanınmış *et al.* (2020) and Tanınmış *et al.* (2022).

Variations of the IM models that are deterministic, referred to as target-set selection problems, have also been studied in the literature, see Chen (2009); Ackerman *et al.* (2010). Raghavan and Zhang (2021a) presented a tight and compact extended formulation for trees and cycle graphs, and further extended these findings to arbitrary networks, having previously implemented a branch-and-cut algorithm in Raghavan and Zhang (2019). Fischetti *et al.* (2018) proposed two exponentially large integer formulations for a model aimed at minimizing the incentives required to influence a fixed proportion of the nodes in a network, and employed cut-generation approaches to solve the proposed models. Günneç *et al.* (2019, 2020) further developed this generalized problem and used branch-and-cut methods to solve it. Two-player versions of these problems have also been studied, such as in Hemmati *et al.* (2014). Recently, Raghavan and Zhang (2021b, 2022) considered the positive influence dominating set problem with and without partial payments and solved these using MIP techniques.

Borrero *et al.* (2018) and Borrero *et al.* (2021) proposed a scalable and general Markovian framework for influence maximization that allows for the modification of network parameters and provides performance upper-bounds on the optimal solution. They also proposed an efficient cut-generation algorithm in which a minimum-cut separation routine is solved to add the corresponding cuts. This cutting plane approach is able to solve many real-life and large instances of the order of 10^4 and 10^5 nodes and arcs.

2.2. Location-aware influence maximization

The Location-aware Influence Maximization Problem (LIMP) is a problem where the goal is to select k initial nodes in a network to maximize the number of influenced nodes that are close to a location of interest. Li *et al.* (2014) showed that the LIMP is *NP*-hard and proposed greedy approximation algorithms to solve it. Wang *et al.* (2016) extended the LIMP by introducing the Distance-aware Influence Maximization Problem, where users have location preferences, and the closer a user is to the query region, the greater its importance. They developed pruning strategies to approximate the solution. Su *et al.* (2018) proposed approximate and heuristic techniques to solve a general version of the LIMP, where a k-size seed set has to be chosen to maximize the expected influence over targeted users who have both topic and location preferences.

Previous models take into account the location of nodes in a network as inputs, but neglect the fact that nodes can move and form trajectories. Furthermore, these studies do not examine how knowing the location of nodes can improve the deployment of ads or Geo-fences. In light of this, Zhang *et al.* (2018) and Li *et al.* (2019) studied the problem of placing billboards. Given a set of trajectories, a set of billboards, and a budget, the goal is to find a subset of billboards that maximizes the number of covered trajectories, where each trajectory is described by a sequence of points. Zhang *et al.* (2018) developed an enumeration-based approximation and a partition-based approximation, whereas Li *et al.* (2019) developed hill-climbing heuristics to solve the problem. However, both studies did not consider the underlying network relationships between nodes.

2.3. Maximum coverage location problems

Maximum Coverage Location Problems (MCLPs) have been widely used in real-world applications, such as the placement of emergency facilities, bus and fire stations, and clustering problems in Geographical Informative Systems (GIS). Examples can be found in Farahani *et al.* (2012), García and Marín (2015), Murray (2016) and other related references therein. Church (1984) introduced the Planar MCLP (PMCLP), where facilities can be placed anywhere on the plane rather than only at a finite set of pre-specified sites. The author defined critical intersection points and showed that there is at least one optimal solution to the PMCLP in which all facilities are in critical intersection points. Murray and Tong (2007) extended the concept of critical intersection points to any polygon representation of demands. Canbolat and von Massow (2009), motivated by wireless transmitter coverage problems, examined the maximal coverage of weighted demand points using axis-parallel ellipses and provided a simulated annealing heuristic to solve the problem. Andretta and Birgin (2013) discussed an exact method and a heuristic for the case when ellipses can be rotated. Recently, Tedeschi and Andretta (2021) proposed exact algorithms for both axis-parallel and freely rotated ellipses using critical points. Bansal and Kianfar (2017) extended the PMCLP under specific assumptions, including rectilinear distance measures, rectangular demand zones, and partial coverage. They developed a heuristic that provides lower bounds for a customized branch-and-bound solution approach.

The aforementioned models prioritize coverage optimization, but none of them take into account the influence of the network in coverage decisions. Our work can be seen as an expansion of PMCLP concepts to settings where there are networked relationships between the elements being covered, and where a complex objective, such as influence, is optimized.

3. Problem formulation

In this section, we describe the ICOP in an arbitrary directed network, prove that it is *NP*-hard, and propose MIP formulations for three different interpretations of coverage. For consistency, we present the problem as placing Geo-fences, which will be referred to as *fences* throughout.

3.1. Definition of ICOP

The present influence model is an adaptation of the model discussed in Borrero *et al.* (2021). Consider a directed network $G=(N,A)$ with $n=|N|$ nodes and $m=|A|$ arcs. The nodes in the network could represent customers, and the arcs could be assumed as social connections between them. A given node can be either active (1) or inactive (0). Influence propagates throughout the network in the following manner: an inactive node $k\in N$ can receive influence from fences that cover it, or become active due to its active in-neighbors $\ell\in N^-(k)$, where $N^-(k)=\{\ell\in N:(\ell,k)\in A\}$ denotes the set of in-neighbors of node k. The time taken for a fence to activate node k is exponentially distributed with the rate δ_k, where $\delta_k\geq 0$ is a constant. The time it takes for an active in-neighbor ℓ to activate node k is exponentially distributed with rate $d_{\ell k}\eta_{\ell k}+g_{\ell k}$, where $d_{\ell k},g_{\ell k}\geq 0$ are constants and the *network rate* $\eta_{\ell k}\geq 0$ is a continuous decision variable. Intuitively, the value of $\eta_{\ell k}$ can be interpreted as a scaled incentive such as a discount or a gift card given to node ℓ for influencing node k. The collection of all η variables is denoted by $\boldsymbol{\eta}=(\eta_{\ell k}:(\ell,k)\in A)$. All activation times are assumed to be independent.

We let F denote the non-empty and finite set of fences and assume that each fence $f\in F$ is represented by a circle of radius $q_f>0$ in the (two-dimensional) plane. We assume that values of q_f and $f\in F$ are fixed and known, and say that a node is *covered* by a fence if it is "sufficiently close" to the center of the fence. Specifically, there are three general models of coverage that we consider:

- Static coverage: Node $k\in N$ has a fixed location represented by $p_k=(p_k^{(1)},p_k^{(2)})$, and is considered covered by fence f if it is within a distance of q_f from the center of the fence, see Figure 2(a).
- Segment coverage: There are two points p_k and $\bar{p}_k$ associated with each node $k\in N$; these points may represent the most frequently visited locations by node k. The segment $L_k=\{v:v=p_k+t(\bar{p}_k-p_k),0\leq t\leq 1\}$, joining p_k and $\bar{p}_k$, is the "average path" taken by node k to move between these two locations. node k is covered by fence $f\in F$ if and only if at least one point of L_k is within a distance of q_f from the center of f, see Figure 2(b).
- Piece-wise coverage: Node $k\in N$ regularly visits $j_k\geq 2$ locations, where p_{kj} are the coordinates of location j, $j\in[j_k]$. The locations are visited sequentially, thus, w.l.o.g, location $j+1$ is visited after j, $j=1,\ldots,j_k-1$. We assume that the path between locations j and $j+1$ is approximated by the segment $L_{kj}=\{v:v=p_{kj}+t(p_{k,j+1}-p_{kj}),0\leq t\leq 1\}$ joining p_{kj} and $p_{k,j+1}$. A fence $f\in F$ covers node k if and only if there is at least one segment L_{kj} within a distance of q_f from the center of the fence, see Figure 3.

For any given coverage mode, the decision variable for the center of fence f is denoted by $\alpha_f=(\alpha_f^{(1)},\alpha_f^{(2)})$, where $\alpha_f^{(1)}$ is the x-coordinate and $\alpha_f^{(2)}$ is the y-coordinate. Also, let $\boldsymbol{\alpha}=(\alpha_f:f\in F)$, and assume that all nodes and possible locations for fences are within a non-empty rectangular region of the plane given by $R=[l_1,u_1]\times[l_2,u_2]$, where $l_i,u_i\in\mathbb{R}$, $i=1,2$, are known and fixed.

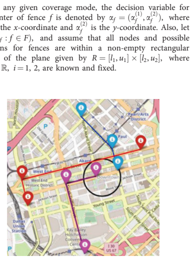
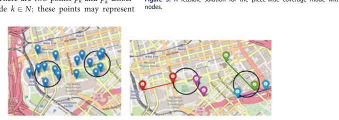


Figure 3. A feasible solution for the piece-wise coverage mode with three nodes.



(a) (b)

Figure 2. (a) A feasible solution for the static coverage mode. The pins represent nodes and the circles are fences; (b) a feasible solution for the segment coverage mode.

For any node $k\in N$, let s_k be equal to 1 if k is active (i.e., influenced) and 0 otherwise. Define $\boldsymbol{s}=(s_k:k\in N)$ as the activation state of the network; at time $t=0$ the network is in some state $\boldsymbol{s}^0\in S_0=\{0,1\}^n$ (we assume hereafter that $\boldsymbol{s}^0=\boldsymbol{0}$ thus initially all nodes are inactive; the general case can be handled after doing simple modifications). Let $X(t,\boldsymbol{\alpha},\boldsymbol{\eta})$ denote the state of the network at time $t\geq 0$ if the fences are located at the coordinates specified by $\boldsymbol{\alpha}$ and the network rates are given by $\boldsymbol{\eta}$. The state space of $X(\boldsymbol{\alpha},\boldsymbol{\eta})=\{X(t,\boldsymbol{\alpha},\boldsymbol{\eta}),t\geq 0\}$ is S_0. From standard theory on Continuous-Time Markov Chains (CTMC) (Kulkarni, 2016), it can be verified that $X(\boldsymbol{\alpha},\boldsymbol{\eta})$ is CTMC over S_0 with an absorbing state $\boldsymbol{1}=(1,\ldots,1)$; in state $\boldsymbol{1}$ all nodes are active.

For any $t\geq 0$, if $X(t,\boldsymbol{\alpha},\boldsymbol{\eta})=\boldsymbol{s}\in S_0$, then $X(\boldsymbol{\alpha},\boldsymbol{s})$ can jump only to states of the form $\boldsymbol{s}+\boldsymbol{e}^k$, where $\boldsymbol{e}^k$ is a unit vector with a one in the kth position and where k is inactive in $\boldsymbol{s}$, that is $s_k=0$. In this case, the time it takes for node k to become active is exponentially distributed with rate $\delta_k n_k(\boldsymbol{\alpha})+\sum_{(\ell,k)\in A}s_\ell(d_{\ell k}\eta_{\ell k}+g_{\ell k})$, where $n_k(\boldsymbol{\alpha})$ is the number of fences that cover node k assuming the fences are located in $\boldsymbol{\alpha}$. By standard CTMC theory (Kulkarni, 2016), the non-diagonal elements of the generator matrix $\Xi(\boldsymbol{\alpha},\boldsymbol{\eta})$ of $X(\boldsymbol{\alpha},\boldsymbol{\eta})$ are given by

$$\Xi_{\boldsymbol{s},\boldsymbol{s}'}(\boldsymbol{\alpha},\boldsymbol{\eta})=\begin{cases}\delta_k n_k(\boldsymbol{\alpha})+\sum_{(\ell,k)\in A}s_\ell(d_{\ell k}\eta_{\ell k}+g_{\ell k}), & \text{if } \boldsymbol{s}'=\boldsymbol{s}+\boldsymbol{e}^k, s_k=0, k\in N\\ 0, & \text{otherwise,}\end{cases}\tag{1}$$

whereas $\Xi_{\boldsymbol{s},\boldsymbol{s}}(\boldsymbol{\alpha},\boldsymbol{\eta})=-\sum_{\boldsymbol{s}'\in S,\boldsymbol{s}'\neq\boldsymbol{s}}\Xi_{\boldsymbol{s},\boldsymbol{s}'}(\boldsymbol{\alpha},\boldsymbol{\eta}),\boldsymbol{s}\in S_0$, for the diagonal elements.

At $t=0$ the decision-maker seeks to determine where the fences should be located and what values should the network influence variables take in order to minimize the time it takes for all nodes to get activated. In other words, the decision-maker determines the values of $\boldsymbol{\alpha}$ and $\boldsymbol{\eta}$ in order to minimize the *Network Activation Time* (NAT). Increasing the value of $\eta_{\ell k}$ by one incurs a cost of $c_{\ell k}$, and the decision-maker has a budget limit of 1, w.l.o.g. Given that the NAT is a random variable, various criteria can be used to optimize it. Following Proposition 3 discussed in Section 3.2 of Borrero *et al.* (2021) (see Appendix VIII for a summary), we frame the minimization of the NAT in terms of maximizing the minimum (non-zero) absolute eigenvalue of the generator of $X(\boldsymbol{\alpha},\boldsymbol{\eta})$ across all its possible values. If $\lambda(\boldsymbol{\alpha},\boldsymbol{\eta})$ denotes this eigenvalue, then we define the ICOP as the following optimization problem

$$\lambda^*=\max\{\lambda(\boldsymbol{\alpha},\boldsymbol{\eta}):\sum_{(\ell,k)\in A}c_{\ell k}\eta_{\ell k}\leq 1,\boldsymbol{\alpha}\in R^{|F|},\boldsymbol{\eta}\in\mathbb{R}_+^m\}.\tag{2}$$

Because the generator matrix $\Xi(\boldsymbol{\alpha},\boldsymbol{\eta})$ of $X(\boldsymbol{\alpha},\boldsymbol{\eta})$ can be rearranged as an upper-triangular matrix, the minimum (non-zero) absolute eigenvalue is equal to the minimum (non-zero) diagonal element in this matrix. That is, $\lambda(\boldsymbol{\alpha},\boldsymbol{\eta})=\min\{|\Xi_{ss}(\boldsymbol{\alpha},\boldsymbol{\eta})|:\boldsymbol{s}\in S_0\setminus\{\boldsymbol{1}\}\}$. The rationale behind problem (2) is that by maximizing $\lambda(\boldsymbol{\alpha},\boldsymbol{\eta})$ one minimizes upper-bounds for both the expected NAT and the tail probabilities of the NAT. Specifically, if $T(\boldsymbol{\alpha},\boldsymbol{\eta})$ denotes the NAT given $\boldsymbol{\alpha}$ and $\boldsymbol{\eta}$, i.e., if $T(\boldsymbol{\alpha},\boldsymbol{\eta})$ is the first passage time of $X(\boldsymbol{\alpha},\boldsymbol{\eta})$ to state $\boldsymbol{1}$, and if $t\geq 0$, then

$$P[T(\boldsymbol{\alpha},\boldsymbol{\eta})>t]\leq Me^{-\lambda(\boldsymbol{\alpha},\boldsymbol{\eta})t}\text{ for } t\geq t_0 \text{ and } E[T(\boldsymbol{\alpha},\boldsymbol{\eta})]\leq\frac{M'}{\lambda(\boldsymbol{\alpha},\boldsymbol{\eta})},\tag{3}$$

where $t_0,M,M'>0$ are some constants; see more details in Appendix VIII.

Table 6 in Appendix I summarizes the notations used throughout the formulations. Next, we show that the ICOP is an *NP*-hard optimization problem for the static coverage mode, implying its *NP*-hardness for all coverage modes discussed in this article. Then, we present a basic MIP formulation of the problem for each coverage mode.

3.2. NP-hardness of ICOP

We prove that the ICOP is *NP*-hard for the static coverage mode by using the planar geometric covering problem, referred to hereafter as the Box Cover Problem (BCP), see Fowler *et al.* (1981). The BCP takes as inputs a collection $\mathcal{C}$ of $m\geq 1$ identical circles with integer radius $r\geq 1$, and a finite set of points $\mathcal{P}$ in the plane with integer coordinates. The problem is to determine if all points in $\mathcal{P}$ can be covered using the circles in $\mathcal{C}$. The BCP is an *NP*-complete problem (Fowler *et al.*, 1981).

Theorem 1. *The ICOP is NP-hard for the static coverage mode.*

3.3. MIP formulations

This section provides MIP formulations for the ICOP in all coverage modes.

3.3.1. Static coverage mode

Let y_{fk} be a binary variable that takes the value 1 if and only if fence f covers node k, $f\in F,k\in N$. Then, for the static coverage mode, problem (2) can be framed as the MIP below:

$$\lambda^*=\max\ \lambda\tag{4a}$$

$$\text{s.t. } \lambda\leq\sum_{k\in N}(1-s_k)(\delta_k\sum_{f\in F}y_{fk}+\sum_{(\ell,k)\in A}s_\ell(d_{\ell k}\eta_{\ell k}+g_{\ell k}))\forall s\in\bar{S}_0\tag{4b}$$

$$D(\alpha_f, p_k) - M_k(1 - y_{fk}) \le q_f \ \forall f \in F; \forall k \in N \quad (4c)$$

$$\sum_{(\ell,k)\in A} c_{\ell k}\eta_{\ell k} \le 1 \quad (4d)$$

$$l_i \le \alpha_f^{(i)} \le u_i \ \forall f \in F; i = 1,2 \quad (4e)$$

$$y_{fk} \in \{0,1\} \forall f \in F; \forall k \in N, \eta_{\ell k} \ge 0 \forall (\ell,k) \in A, \lambda \ge 0. \quad (4f)$$

Given that $\bar{S}_0$ is defined as $\bar{S}_0 = S_0 \setminus \{\mathbf{1}\}$, constraints (4b) ensure that λ is equal to the absolute value of the smallest eigenvalue of the generator matrix of $X(\boldsymbol{\alpha}, \boldsymbol{\eta})$, which is being maximized in problem (2). Inequality (4c) sets the condition that fence f can cover node k if the Euclidean distance between them, represented as $D(\alpha_f, p_k)$, is no greater than the coverage radius of fence f, denoted as q_f. The coverage region of a fence is modeled as a circle and thus, $D(\alpha_f, p_k) = ||\alpha_f - p_k||$, where $||x||$ represents the Euclidean norm of vector x. Note that the constraint includes the "big-M" constant M_k that is computed as $M_k = \max\{||p_k - v_{ij}|| : i,j = 1,2\}$, v_{ij} being the two-dimensional vector $[l_i u_j]$. Inequalities (4d) and (4e) represent the constraints for the budget allocated to increase the network influence rates and the allowed region for placing the fences, respectively.

3.3.2. Segment coverage mode

In this mode, we replace constraint (4c) in Formulation (4) by the following constraint:

$$D(\alpha_f, L_k) - M_k(1 - y_{fk}) \le q_f \ \forall f \in F; \forall k \in N, \quad (5)$$

where M_k is a sufficiently large value that will be calculated in Section 4, and $D(\alpha_f, L_k)$ denotes the Euclidean distance between α_f and the segment L_k.

Let C_f be the circle of radius q_f with center α_f representing fence f, and observe that fence f covers L_k if and only if $C_f \cap L_k \neq \emptyset$. Because $L_k = \{v \in \mathbb{R}^2 : v = p_k + t(\bar{p}_k - p_k), 0 \le t \le 1\}$ we have that $C_f \cap L_k \neq \emptyset$ if and only if there exists $t_{fk} \in [0,1]$ such that $||p_k + t_{fk}(\bar{p}_k - p_k) - \alpha_f|| \le q_f$. The MIP formulation, which we refer to hereafter as the SST, is derived from Formulation (4) by adding new variables t_{fk} for all $f \in F$ and $k \in N$, removing constraints (4c), and including the constraints

$$||p_k + t_{fk}(\bar{p}_k - p_k) - \alpha_f|| - M_k(1 - y_{fk}) \le q_f \ \forall f \in F; \forall k \in N \quad (6a)$$

$$0 \le t_{fk} \le 1 \ \forall f \in F; \ \forall k \in N. \quad (6b)$$

The formulation for piece-wise coverage mode (PWT) is provided in Appendix IV.

4. Formulation enhancements

In this section, we discuss two sets of valid inequalities, a reformulation for the non-linear "big-M" constraints, and heuristic approaches. The improvements are then leveraged in the proposed IDBC algorithm, which will be presented in the next section. The discussion in this section is mainly focused on the SST formulation, i.e., the segment coverage mode, but the results can be easily generalized to the other two modes, the Static and Piece-wise modes (see Appendix IV).

4.1. Conflict inequalities

These conflict-type inequalities are built upon the idea that if two segments are distant enough, there can be no fence that covers both of them. For given nodes $k, \ell \in N$, let $\Lambda_{k\ell} = \{f \in F : D(L_k, L_\ell) > 2q_f\}$ denote the set of fences that cannot cover both L_k and L_ℓ, where $D(L_k, L_\ell)$ measures the Euclidean distance between segments L_k and L_ℓ. Note that $D(L_k, L_\ell) = \min_{t_k, t_\ell}\{||p_k + t_k(\bar{p}_k - p_k) - p_\ell - t_\ell(\bar{p}_\ell - p_\ell)|| : t_k, t_\ell \in [0,1]\}$. It is clear that the following constraints, which we refer to hereafter as the *conflict inequalities*, are valid for the SST:

$$y_{fk} + y_{f\ell} \le 1 \quad \forall k, \ell \in N; \forall f \in \Lambda_{k\ell}. \quad (7)$$

We need to compute $D(L_k, L_\ell)$ to create set $\Lambda_{k\ell}$. To this end, we have to solve a minimization problem over t_k and t_ℓ, which can be solved by finding the *stationary points*. Specifically, instead of the norm, we can minimize the squared-norm function $S(t_k, t_\ell) = ||p_k + t_k(\bar{p}_k - p_k) - p_\ell - t_\ell(\bar{p}_\ell - p_\ell)||^2$, which is "easy" to optimize as implied next. Using Remark 1, one can include all $O(n^2|F|)$ conflict constraints into the SST in $O(n^2|F|)$ time.

Remark 1. $S(t_k, t_\ell)$ is a convex function of $(t_k, t_\ell) \in \mathbb{R}^2$. Thus, by taking the partial derivatives of $S(t_k, t_\ell)$ with respect to t_k and t_ℓ, and making them equal to zero, we have a single stationary point of $S(t_k, t_\ell)$ satisfying

$$t_k^* = \frac{b'}{b}t_\ell - \frac{c'}{b} \quad \text{and} \quad t_\ell^* = \frac{cb + c'a}{b^2 + b'a}. \quad (8)$$

Where the coefficients a, b, b', c are given by $a = -2[(\bar{p}_k^{(1)} - p_k^{(1)})^2 + (\bar{p}_k^{(2)} - p_k^{(2)})^2]$; $b = 2[(\bar{p}_k^{(1)} - p_k^{(1)})(\bar{p}_\ell^{(1)} - p_\ell^{(1)}) + (\bar{p}_k^{(2)} - p_k^{(2)})(\bar{p}_\ell^{(2)} - p_\ell^{(2)})]$; $c = 2[(\bar{p}_k^{(1)} - p_k^{(1)})(p_k^{(1)} - p_\ell^{(1)}) + (\bar{p}_k^{(2)} - p_k^{(2)})(p_k^{(2)} - p_\ell^{(2)})]$; $b' = -2[(\bar{p}_\ell^{(1)} - p_\ell^{(1)})^2 + (\bar{p}_\ell^{(2)} - p_\ell^{(2)})^2]$; and $c' = 2[(\bar{p}_\ell^{(1)} - p_\ell^{(1)})(p_k^{(1)} - p_\ell^{(1)}) + (\bar{p}_\ell^{(2)} - p_\ell^{(2)})(p_k^{(2)} - p_\ell^{(2)})]$. The convexity of $S(t_k, t_\ell)$ now implies that the values of (t_k, t_ℓ) attaining the minimum of $S(t_k, t_\ell)$ are given by (8), as long as they both belong to [0,1]. Else, we have to consider the value of $S(t_k, t_\ell)$ when t_k, t_ℓ are either zero or one. Specifically, suppose for any $k, \ell \in N$, the pair (t_k^*, t_ℓ^*) is given by (8). Then, $D^2(L_k, L_\ell) = S(\hat{t}_k, \hat{t}_\ell)$ where $(\hat{t}_k, \hat{t}_\ell) = (t_k^*, t_\ell^*)$ if $0 \le t_k^*, t_\ell^* \le 1$. Else, $(\hat{t}_k, \hat{t}_\ell)$ can be computed using Table 1.

Table 1. Values of $(\hat{t}_k, \hat{t}_\ell)$ for Remark 1.

Condition	*Value of* $(\hat{t}_k, \hat{t}_\ell)$	*Condition*	*Value of* $(\hat{t}_k, \hat{t}_\ell)$
$t_k^*, t_\ell^* < 0$	(0,0)	$c/a \in [0,1], t_\ell^* < 0$	$(c/a, 0)$
$t_k^* < 0, t_\ell^* > 1$	(0,1)	$t_k^* < 0, c'/b' \in [0,1]$	$(0, c'/b')$
$t_k^* > 1, t_\ell^* < 0$	(1,0)	$(c - b)/a \in [0,1], t_\ell^* > 1$	$((c - b)/a, 1)$
$t_k^* > 1, t_\ell^* > 1$	(1,1)	$t_k^* > 1, (c' + b)/b' \in [0,1]$	$(1, (c' + b)/b')$

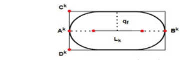


Figure 4. Capsule and rectangle around L_k. A^k and B^k are the intersection of the extension of L_k with the enveloping rectangle.

4.2. Rectangular inequalities

We know that fence $f \in F$ covers segment L_k if and only if α_f belongs to the region $\{v \in \mathbb{R}^2 : D(v, L_k) \le q_f\}$. Such a region resembles a *capsule* in the plane, see Figure 4. Capsules cannot be described by linear terms, making it difficult for standard MIP solvers to handle. As a solution, a rectangular region that envelops the capsule, as shown in Figure 4, can be used as an approximation. The derivation of these regions is provided in the next result; its proof is in Appendix II.

Proposition 1. *Let $f \in F, k \in N$, and $\alpha_f \in \mathbb{R}^2$ be given. If $D(\alpha_f, L_k) \le q_f$ then α_f satisfies that*

$$0 \le \frac{(\alpha_f - A_k)^\top (B_k - A_k)}{||B_k - A_k||^2} \le 1 \text{ and } 0 \le \frac{(\alpha_f - C_k)^\top (D_k - C_k)}{||D_k - C_k||^2} \le 1,$$

where points $A_k = (A_k^{(1)}, A_k^{(2)})$, $B_k = (B_k^{(1)}, B_k^{(2)})$, $C_k = (C_k^{(1)}, C_k^{(2)})$, and $D_k = (D_k^{(1)}, D_k^{(2)})$, are given by

$$A_k = p_k - \frac{q_f}{D(p_k, \bar{p}_k)}(\bar{p}^k - p_k),$$

$$B_k = p_k + \left(1 + \frac{q_f}{D(p_k, \bar{p}_k)}\right)(\bar{p}_k - p_k),$$

$C_k^{(1)} = A_k^{(2)} - p_k^{(2)} + A_k^{(1)}$, $C_k^{(2)} = p_k^{(1)} - A_k^{(1)} + A_k^{(2)}$, $D_k^{(1)} = p_k^{(2)} - A_k^{(2)} + A_k^{(1)}$, *and* $D_k^{(2)} = A_k^{(1)} - p_k^{(1)} + A_k^{(2)}$.

Proposition 1 implies that the following constraints are valid for the SST:

$$\frac{(\alpha_f - A_k)^\top (B_k - A_k)}{||B_k - A_k||^2} - l_{ab}^k(1 - y_{fk}) \ge 0 \ \forall f \in F; \forall k \in N \quad (9a)$$

$$\frac{(\alpha_f - A_k)^\top (B_k - A_k)}{||B_k - A_k||^2} - u_{ab}^k(1 - y_{fk}) \le 1 \ \forall f \in F; \forall k \in N \quad (9b)$$

$$\frac{(\alpha_f - C_k)^\top (D_k - C_k)}{||D_k - C_k||^2} - l_{cd}^k(1 - y_{fk}) \ge 0 \quad \forall f \in F; \forall k \in N \quad (9c)$$

$$\frac{(\alpha_f - C_k)^\top (D_k - C_k)}{||D_k - C_k||^2} - u_{cd}^k(1 - y_{fk}) \le 1 \quad \forall f \in F; \forall k \in N, \quad (9d)$$

where $l_{ab}^k, l_{cd}^k, u_{ab}^k$, and u_{cd}^k are "big-M" constants for each $k \in N$.

Although the constraints (9) include "big-M" constants, they can be computed easily and have a similar magnitude to the remaining non-zeros in the corresponding rectangular inequalities. Consequently, the "big-M"s are not expected to lead to loose relaxation bounds or poor performance, as is a common case in formulations with "big-Ms". Indeed, consider constraint (9a) and note that l_{ab}^k can be computed as $l_{ab}^k = \min\{\alpha^\top (B_k - A_k)/||B_k - A_k||^2 : \alpha \in \{(l_1 l_2), (l_1 u_2), (u_1 l_2), (u_1 u_2)\}\}$, which boils down to comparing four numbers. A similar approach applies to compute u_{ab}^k, l_{cd}^k, and u_{cd}^k.

4.3. Reformulation of the non-linear big-M constraints

The MIP formulation for the SST contains the non-linear "big-M" constraint (6a). Dealing with this constraint directly leads to severe numerical issues, particularly because it is necessary to consider its squared form to be directly fed into a quadratic optimization solver with mixed-integer optimization functionality such as CPLEX (IBM ILOG, 2021) or Gurobi (Gurobi Optimization, LLC, 2021). This squared form is

$$||p_k + t_{fk}(\bar{p}_k - p_k) - \alpha_f||^2 - \hat{M}_k(1 - y_{fk}) \le q_f^2 \quad \forall f \in F; \forall k \in N, \quad (10)$$

where $\hat{M}_k$ is an upper bound on the squared distance $||p_k + t_{fk}(\bar{p}_k - p_k) - \alpha_f||^2$. Next, we propose an alternative quadratic reformulation of this constraint that allows the "big-M" to be an upper-bound on the actual distance rather than the squared distance, and that avoids having the "big-M" in the quadratic constraint. We note that (10) can be reformulated in other ways. In Appendix VII we provide other reformulations (one of them following Atamtürk and Narayanan (2010)) and show that all the considered reformulations have similar performance within our algorithm.

Proposition 2. *Let SST' be obtained from the SST by replacing constraint (6a) with:*

$$(v_{fk1})^2 + (v_{fk2})^2 \le q_f^2 \quad \forall f \in F; \quad \forall k \in N \quad (11a)$$

$$v_{fki} \le p_k^{(i)} + t_{fk}(\bar{p}_k^{(i)} - p_k^{(i)}) - \alpha_f^{(i)} + M_{ki}(1 - y_{fk}) \quad \forall f \in F; \forall k \in N; i = 1,2 \quad (11b)$$

$$v_{fki} \ge p_k^{(i)} + t_{fk}(\bar{p}_k^{(i)} - p_k^{(i)}) - \alpha_f^{(i)} - M_{ki}(1 - y_{fk}) \quad \forall f \in F; \forall k \in N; i = 1,2, \quad (11c)$$

where v_{fki}, $f \in F, k \in N$, and $i = 1, 2$ are continuous variables. If the solution $(\lambda', y', t', \alpha', v')$ is considered optimal in the SST', for $v'_{fki}, f \in F, k \in N$, and $i = 1, 2$, then there exists an equivalent optimal solution (λ, y, t, α) in the SST. Moreover, the optimal values of both the SST and the SST' are equal.

Observe that the "big-M" values M_{ki} in (11) can be tightened further than $u_i - l_i$. In this regard, consider the segment L_k as shown in Figure 5, then it is clear that for $f \in F; k \in N$ and $i = 1, 2$, we can set $M_{ki} = \max\{(u_i - p_k^{(i)}), (l_i - p_k^{(i)}), (u_i - \bar{p}_k^{(i)}), (l_i - \bar{p}_k^{(i)})\}$.

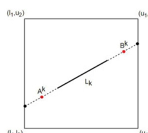


Figure 5. Intersection of segment $A_k B_k$ with vertical sides.

4.4. Heuristic enhancements

This section discusses a heuristic method for finding an initial feasible solution for the SST, based on the *k-means* clustering algorithm (Likas *et al.*, 2003). This method divides nodes into $|F|$ groups based on the distance between each node and the center of the fences. The result is a binary vector y^0, which is as a starting point for finding a feasible solution in the MIP formulation.

Heuristic 1 in Appendix III adapts the *k*-means algorithm to our setting. It starts by randomly placing each fence on the plane, and assigns segments to the nearest fence. Then, in each iteration, the fence locations are updated to be the average of the assigned segments' coordinates, and segments are re-assigned to the closest fence. This is repeated for U iterations, where U is a parameter set by the user.

The solution from Heuristic 1 is further improved as follows. Let $R_f^0 = \{r^*k : b_{fk} = 1\}$ represent the coordinates of the points covered by fence f in the heuristic solution. For each fence $f \in F$, the improvement process checks the number of nodes covered when the fence location is changed to each of the points in R_f^0. The location that covers the most nodes is then chosen as the new coordinate for fence f. If multiple radii values are possible, the procedure considers different radius assignments, as described in Heuristic 1 in the Appendix III.

Note that both the heuristic and polishing steps can be performed in polynomial time if all fence radii are the same. If the radii are different, then $O(|F|!)$ operations would need to be considered when assigning radii to the final coordinate assignments. However, this is not expected to be a problem as the number of fences $|F|$ is usually small (e.g., less than 10).

5. IDBC algorithm

This section explains a procedure to solve the formulation SST using an algorithm IDBC. The algorithm begins with a master relaxation of the formulation and adds two types of constraints: the state constraints (4b) and the reformulated "big-M" constraints (11). Section 5.1 provides an overview of the algorithm whereas Section 5.2 provides more details into the Decomposition Branch-and-Cut (DBC).

5.1. Overview of the IDBC algorithm

The IDBC begins by constructing a master relaxation of the SST that includes the conflict and rectangular inequalities. This relaxation only includes a portion of the *state* constraints (i.e., those specified by (4b)) and does not include any "big-M" constraints (i.e., those specified by (6)). The algorithm then adds state constraints on the fly by solving a min-cut optimization problem, which separates optimal master solutions that are infeasible in the SST. When no additional state constraints are required, the algorithm checks if the integer optimal solution of the master satisfies all "big-M" constraints of the form (6). If it does, an optimal solution for the SST has been found. If not, new variables and constraints of the form given by (11) are added to the master, and the resulting MIP is *resolved* "from scratch" using the DBC algorithm in the next iteration.

In order to describe the IDBC in more detail, let a subset of nodes $V \subseteq N$ and a subset of states $T \subseteq \bar{S}_0$ be given. Denote by $\xi(\boldsymbol{s}, \boldsymbol{y}, \boldsymbol{\eta})$ the right-hand side function in constraint (4b) and consider the MIP

$$\Lambda(T, V) = \max \ \lambda \quad (12a)$$

$$\text{s.t.} \ \lambda \le \xi(\boldsymbol{s}, \boldsymbol{y}, \boldsymbol{\eta}) \quad \forall \boldsymbol{s} \in T \quad (12b)$$

$$(v_{fk1})^2 + (v_{fk2})^2 \le q_f^2 \quad \forall f \in F; \forall k \in V \quad (12c)$$

$$v_{fki} \le p_k^{(i)} + t_{fk}(\bar{p}_k^{(i)} - p_k^{(i)}) - \alpha_f^{(i)} + M_{ki}(1 - y_{fk}) \quad \forall f \in F; \forall k \in V; i = 1,2 \quad (12d)$$

$$v_{fki} \ge p_k^{(i)} + t_{fk}(\bar{p}_k^{(i)} - p_k^{(i)}) - \alpha_f^{(i)} - M_{ki}(1 - y_{fk}) \quad \forall f \in F; \forall k \in V; i = 1,2 \quad (12e)$$

$$\sum_{(\ell,k)\in A} c_{\ell k}\eta_{\ell k} \le 1 \quad (12f)$$

$$\text{Constraints (7) and (9)} \quad (12g)$$

$$y \in \{0,1\}^{n|F|}, \eta \in \mathbb{R}_+^m, \boldsymbol{\alpha} \in R^{2|F|}, t \in [0,1]^{n|F|}, v \in \mathbb{R}_+^{2n|F|}, \lambda \in \mathbb{R}. \quad (12h)$$

Observe that $\Lambda(T, V) \ge \lambda^*$ because (12) is a relaxation of the SST and note that the conflict and rectangular constraints are added for all nodes in N and not just those in V.

The detailed IDBC procedure is shown in Algorithm 1. Clearly λ^* and ζ^* are the optimal value and an optimal solution of the SST, respectively, because ζ^* is feasible in the SST and is optimal in a relaxation of the SST that has the same objective function. Algorithm 1 terminates in at most n iterations. Additionally, each iteration is guaranteed to terminate (see Section 5.2), ensuring that the entire algorithm terminates. In practice, however, the algorithm terminates in far fewer iterations than the theoretical worst-case.

Algorithm 1: IDBC algorithm to find ζ^* and λ^* for SST.

Input : Node set N, arc set A, initial set of states $T^0 \subseteq \bar{S}_0$.

Output: An optimal solution ζ^* and λ^*.

1 Put $V^0 = \emptyset$, $i = 0$, $\zeta^0 =$ null;

2 Obtain y^0 from Heuristic 1 and use it to initialize $\Lambda(T^0; V^0)$;

3 **while** ζ^i *is not feasible in SST* **do**

4 $\quad i = i + 1$;

5 $\quad$ Solve $\lambda^i = \Lambda(\bar{S}_0, V^i)$ using the DBC starting with the states in T^i (see Section 5.2). Let $\bar{T}^i$ be the set of states added by the DBC and let ζ^i be the optimal solution attaining λ^i;

6 $\quad$ **if** ζ^i *does not satisfy all (4c) constraints* **then**

7 $\quad\quad$ Let $N' \subseteq N$ be the nodes where (4c) is not satisfied, set $V^{i+1} = V^i \cup N'$, and $T^{i+1} = T^i \cup \bar{T}^i$;

8 $\quad$ **end**

9 **end**

10 **return** ζ^i and λ^i.

5.2. A DBC algorithm to solve $\Lambda(\bar{S}_0, V^i)$

In step 5 of Algorithm 1, the MIP $\Lambda(\bar{S}_0, V^i), i \ge 1$, is solved using a DBC algorithm. To this end, $\Lambda(T^i, V^i)$ is used as a master problem and cuts are added on the fly. To add cuts, a "separation problem" is defined in a similar way to how Algorithm 1 of Borrero *et al.* (2021) defines cuts in their cutting-generation framework. In Borrero *et al.* (2021), cuts are iteratively added to a linear program whereas here we need to implement them as "lazy cuts" deployed in a DBC algorithm. Next, we discuss how the cutting plane method of Borrero *et al.* (2021) adapts to the proposed DBC algorithm.

Consider a feasible candidate solution $\zeta = (\lambda, \eta, y, \alpha, t, v)$ of $\Lambda(T^i, V^i)$ and define the auxiliary network $\mathcal{G}(\zeta) = (N', A')$, where

$$N' = N \cup \{\rho_0\} \text{ and } A' = A \cup \{(\rho_0, \ell) : \ell \in N\}). \quad (13)$$

In (13), ρ_0 is defined as an auxiliary "root" node that can be thought of as representing all the fences. The weights of the arcs in $\mathcal{G}(\zeta)$ are defined by

$$w_{jk}(\zeta) = \begin{cases} \delta_k \sum_{f \in F} y_{fk}, & \text{if } j = \rho_0 \text{ and } k \neq \rho_0 \\ d_{jk}\eta_{jk} + g_{jk}, & \text{if } j \neq \rho_0 \text{ and } k \neq \rho_0. \end{cases} \quad (14)$$

for any $(j, k) \in A'$. Given any $H \subseteq N'$ with $\rho_0 \in H$, we refer to $(H, N' \setminus H)$ as a *rooted-out* cut, and define its weight by $w(H, \zeta) = \sum_{j \in H, k \in N' \setminus H} w_{jk}(\zeta)$.

Define $H_{\boldsymbol{s}} = \{k \in N : s_k = 1\} \cup \{\rho_0\}$, then the right-hand side value $\xi(\boldsymbol{s}, \boldsymbol{y}, \boldsymbol{\eta})$ becomes equal to $w(H_{\boldsymbol{s}}, \zeta)$, for any given $\boldsymbol{s} \in \bar{S}_0$. On the other hand, it is readily seen that $\{H_{\boldsymbol{s}} : \boldsymbol{s} \in \bar{S}_0\} = \{H \subseteq N' : \rho_0 \in H\}$. Applying these definitions, we can rewrite $\Lambda(\bar{S}_0, V)$ as the following max-min-cut problem over $\mathcal{G}(\zeta)$:

$$\Lambda(\bar{S}_0, V) = \max\{\min\{w(H, \zeta) : H \subseteq N', \rho_0 \in H\} : \zeta \in \mathcal{B}\}, \quad (15)$$

where $\mathcal{B}$ is the feasible set for values of ζ, i.e., it contains all constraints (12c) to (12h). Therefore, we have the following result that relates the feasibility of ζ in $\Lambda(\bar{S}_0, V^i)$ with the value of the (rooted-out) min-cut problem in $\mathcal{G}(\zeta)$.

Proposition 3. *Let $T \subseteq \bar{S}_0$ and $V \subseteq N$ be given, and assume that $\zeta = (\lambda, \eta, y, \alpha, t, v)$ is a feasible solution of $\Lambda(T, V)$. Then, ζ is feasible in $\Lambda(\bar{S}_0, V)$ if and only if $\lambda \le \min\{w(H, \zeta) : H \subseteq N', \rho_0 \in H\}$.*

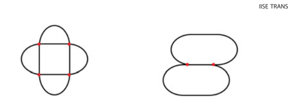
Using Proposition 3 we solve $\Lambda(\bar{S}_0, V^i), i \ge 1$, with a DBC as follows. Initially, when $U^i = \emptyset$ the linear relaxation of $\Lambda(T^i, V^i)$ is solved at the root node of the Branch-and-Bound (BnB) tree, and branching proceeds on the binary variables y as usual. If in a node of the BnB tree an integer feasible solution $\hat{\zeta} = (\hat{\lambda}, \hat{\eta}, \hat{y}, \hat{\alpha}, \hat{t}, \hat{v})$ of $\Lambda(T^i, V^i)$ is found (i.e., a feasible solution that satisfies all the integrality requirements on $\hat{y}$), then the rooted-out min-cut problem $\hat{\omega} = \min\{w(H, \hat{\zeta}) : H \subseteq N', \rho_0 \in H\}$ is solved. If $\hat{\lambda} \le \hat{\omega}$, then, by Proposition 3, $\hat{\zeta}$ is a feasible solution of $\Lambda(\bar{S}_0, V^i)$ and the current node of the BnB tree is pruned by feasibility. Else, there exists a state $\hat{\boldsymbol{s}} \in \bar{S}_0 \setminus S^i$ such that $\hat{\lambda} > \xi(\hat{\boldsymbol{s}}, \hat{y}, \hat{\eta})$. Indeed, let $(\hat{H}, N' \setminus \hat{H})$ be the cut attaining $\hat{w}$ and define $\hat{\boldsymbol{s}} = (\hat{s}_k : k \in N)$ by $\hat{s}_k = 1$ if $k \in \hat{H}$, and zero otherwise. Then, it is readily verified that $\hat{w} = \xi(\hat{\boldsymbol{s}}, \hat{y}, \hat{\eta})$, which gives the desired claim (note that it must be the case that $\hat{\boldsymbol{s}} \notin S^i$, because if not there would be a contradiction with the feasibility of $\hat{\zeta}$). Once the state $\hat{s}$ is identified, we add the constraint $\lambda \le \xi(\hat{s}, y, \eta)$ to the formulation $\Lambda(T^i, V^i)$ at the current node, the node is resolved, and we set $U^i = U^i \cup \{\hat{s}\}$.

We note that the proposed DBC's integer separation process can be completed in polynomial time, specifically in $O(n^3 m \log(n^2/m))$, through the use of the Hao–Orlin algorithm (Hao and Orlin, 1992). Furthermore, $\Lambda(\bar{S}_0, \emptyset)$ is polynomially solvable. This is because if we fix $y_{fk} = 1$ for all $f \in F$ and $k \in N$, it reduces $\Lambda(\bar{S}_0, \emptyset)$ to a NAT problem, which is polynomially solvable, as shown in Borrero *et al.* (2021). The complexity of solving $\Lambda(\bar{S}_0, T^i)$ increases as T^i increases. Thus, the initial iterations of the IDBC are usually fast, whereas later iterations are more time-consuming. Lastly, the DBC to solve $\Lambda(\bar{S}_0, V)$ finds the optimal solution after adding at most $|\bar{S}^0 \setminus T|$ cuts, which ensures the DBC terminates.

Remark 2. An important consideration in the algorithmic design to solve the SST is about whether to solve a single MIP problem with a single run of a DBC algorithm (a "unified" DBC approach), or to iteratively solve several MIPs in the way it is done in Algorithm 1. In contrast with the iterative approach, in the unified DBC we would add the constraints (11) for all nodes in $k \in N'$ on the fly after step 7, then re-solve the node, and then continue the BnB process. Preliminary experiments with the instances of Section 7 indicated that the IDBC approach generally outperforms the "unified" DBC approach. This is believed to be due to the use of the (11)-type constraints in all nodes of the BnB tree in the IDBC, which results in a tighter master relaxation compared with the unified DBC.

6. Discrete formulation of the ICOP

Next, we present a discrete formulation for the ICOP based on intersection points. Church (1984) proved that for the Planar Maximal Covering Problem under the Euclidean distance (PMCE), there is at least one optimal solution in which all facilities are located at points that belong to the *Circle Intersection Point set* (CIPS). This result generalizes to many other planar settings, including to the ICOP, as discussed next. However, the formulation uses a quadratic number of binary variables in terms of the number of nodes, which hinders its practicality. Below, we explain the main ideas in terms of the segment coverage mode. The results can also be generalized to static and piece-wise modes.



(a) Four critical points. (b) Two critical points.

Figure 6. Two scenarios of capsule intersection.

Let $f \in F$ be fixed, and recall that L_k, $k \in N$, denotes the segment of the kth node with endpoints p_k and $\bar{p}_k$. We define C_k^f as the border of the capsule around L_k with a radius of q_f, as shown in Figure 6. For a given $\ell, k \in N$, let $I_{\ell k}^f$ define the intersection of C_ℓ^f and C_k^f. If the intersection is finite (as shown in Figure 6(a)), $I_{\ell k}^f$ represents the actual intersection points. In case where the intersection has infinitely many points (as depicted in Figure 6(b)), $I_{\ell k}^f$ represents the endpoints of the intersection.

Let $\mathcal{I}^f$ be the intersection point set associated with fence f, which is defined by

$$\mathcal{I}^f = \bigcup_{k \in N}(\{p_k, \bar{p}_k\}) \cup \bigcup_{\ell \subseteq N} I_{\ell k}^f). \quad (16)$$

In other words, $\mathcal{I}^f$ contains all the endpoints of the L_k segments, plus the pairwise intersection points between all the capsules C_ℓ^f and C_k^f. The *Intersection Point Set* (IPS) is defined as $\mathcal{I} = \cup_{f \in F}\mathcal{I}^f$. Observe that if all the fences have the same radius, then $\mathcal{I} = \mathcal{I}^f$. Moreover, $\mathcal{I}$ contains $O(|F||N|^2)$ elements. Proposition 4 states that $\mathcal{I}$ contains at least an optimal solution for the ICOP; see Appendix II for the proof.

Proposition 4. *There is at least one optimal solution to the ICOP in that all fence locations belong to $\mathcal{I}$.*

Let $\mathcal{F} \subseteq F$ be the largest subset of F in which no two fences have the same radius. For each $k \in N$ and $f \in \mathcal{F}$, let $\Delta_{kf} \subseteq \mathcal{I}^f$ be the set of intersection points in $\mathcal{I}^f$ obtained from capsules with radius q_f that can cover L_k; specifically, $\Delta_{kf} = \{p_k, \bar{p}_k\} \cup \bigcup_{\ell \in N} I_{\ell k}^f$. Also, we define h_f as the number of fences with radius q_f required to be located. We introduce the binary variable z_{if}, which is equal to 1 if a fence is located at intersection point $i \in \mathcal{I}^f$, and 0 otherwise. The discrete formulation for ICOP, hereafter referred to as D-ICOP, is:

$$\hat{\lambda}^* = \max \lambda \quad (17a)$$

$$\text{s.t.} \ \lambda \le \sum_{k \in N}(1 - s_k)(\delta_k \sum_{f \in \mathcal{F}}\sum_{i \in \Delta_{kf}} z_{if} + \sum_{(\ell,k)\in A} s_\ell(d_{\ell k}\eta_{\ell k} + g_{\ell k})) \quad \forall s \in \bar{S}_0 \quad (17b)$$

$$\sum_{i \in \mathcal{I}^f} z_{if} \le h_f \quad \forall f \in \mathcal{F} \quad (17c)$$

$$\sum_{(\ell,k)\in A} c_{\ell k}\eta_{\ell k} \le 1 \quad (17d)$$

$$z_{if} \in \{0,1\} \forall i \in \mathcal{I}^f; f \in \mathcal{F}; \eta_{\ell k} \ge 0 \forall (\ell, k) \in A, \lambda \ge 0. \quad (17e)$$

D-ICOP offers a key advantage over the original formulation (4) by avoiding the inclusion of non-linear constraints and the need for "big-M" constants. However, it brings in certain drawbacks: First, it requires a precomputation of all the sets Δ_{kf}; second, it could end up having $O(|F||N|^2)$ binary variables (because there is a binary variable in (17) for each intersection point in $\mathcal{I}$).

In order to decrease the required number of binary variables we can apply a preprocessing step. Let us fix $f \in \mathcal{F}$, and define Φ_i as the set of segments covered by the intersection point i, $i \in \mathcal{I}^f$. If it is found that $\Phi_i \subseteq \Phi_j$ for a pair $i, j \in \mathcal{I}^f$, then we can remove i from $\mathcal{I}^f$. Whereas the complexity of such a naive approach is $O(|\mathcal{I}^f|^2)$ for each $f \in \mathcal{F}$, the reduction in the number of binary variables can be significant enough to improve the computational efficiency for solving the D-ICOP.

7. Computational experiments

In this section, numerical experiments are conducted to analyze the sensitivity of the proposed model to different parameter configurations, determine the insights on the influence process that the model can provide, and compare the performance of Algorithm 1 to solve the problem versus simpler cutting plane algorithms and the D-ICOP formulation. To see the effects of enhancements in the IDBC algorithm, and comparisons with the maximum coverage problem, refer to Appendix IX. The focus is on the segment covering problem, i.e., the SST formulation, with the static coverage and piece-wise coverage models being discussed in Appendix VI.

All instances are solved using CPLEX 12.9.0 and the min-cut problems are solved by implementing the Hao–Orlin algorithm in the LEMON graph library (Dezső *et al.*, 2011). The algorithm is coded in C++ and the experiments are carried out on a 64-bit Windows 10 Pro laptop with 8GB of RAM and an eighth-generation i7 1.8GHz processor.

7.1. Description of the instances

We consider three set of experiments with different parameter configurations to analyze different types of underlying social interactions:

- **External factor set**: We have $d_{\ell k} = 0; \forall (\ell, k) \in A$. Intuitively, the decision-maker cannot change the network influence rates. However, the underlying social interactions still play a role in the objective function of the ICOP since $g_{\ell k} > 0; \forall (\ell, k) \in A$.
- **General set**: We have $d_{\ell k}, g_{\ell k} > 0; \forall (\ell, k) \in A$, meaning that the decision-maker is able to modify the network influence rates by investing the budget.
- **Average set**: All $d_{\ell k}, g_{\ell k} > 0; \forall (\ell, k) \in A$; and $d_{\ell k}$s are equal to the average of the corresponding values in the general set.

We consider two classes of network topologies across 10 instances as shown in Table 2. Six of them are directed versions of networks following the Watts–Strogatz (WS) preferential attachment model (Watts and Strogatz, 1998), and the remaining four are real instances from the SNAP repository (Leskovec and Krevl, 2014). The rest of the parameters defining the instances and the algorithm are described in Appendix V.

Table 2. Description of instances in terms of number of nodes, number of arcs, and density.

Instance name	Nodes (n)	Arcs (m)	Density
20.4.3	20	80	0.21
30.4.3	30	120	0.14
40.4.3	40	160	0.10
50.4.3	50	200	0.08
100.4.3	100	400	0.04
120.4.3	120	480	0.03
140.4.3	140	560	0.03
karate	34	156	0.14
dolphins	62	318	0.08
polbooks	105	882	0.08
football	115	1226	0.09

7.2. Results and discussion

For each combination of parameter sets and network topologies, we randomly generate three instances. We report the average results of these three replications in Tables 3 to 5. These tables present the results of experiments with two fences and coverage radii of one and two. In all tables, the solution times are reported in seconds. The "OBJ" column represents the optimal solution, whereas the "HeurOBJ" column represents the value of the feasible solution found by Heuristic 1. The "UC" column stands for the number of lazy cuts, and "TUC" represents the total number of lazy cuts, including those added by the solver when proving the feasibility of the heuristic solution. The "Cov" column shows the number of covered nodes, "#M" stands for the total number of violated "big-M" constraints, and "#Re_MIPs"

Table 3. Results of the external factor set for two fences with radius 1 and 2.

	IDBC								D-ICOP		
Instance	*Time*	*OBJ*	*HeurOBJ*	*UC*	*TUC*	*Cov*	*#M*	*#Re_MIPs*	*IPS*	*PrePro*	*Formu*
20.4.3.0	0.08	21.00	15.33	0	0	9	4	1	0.50	0.09	0.04
30.4.3.0	0.14	14.44	10.89	0	0	9	4	1	1.46	0.65	0.06
40.4.3.0	0.22	9.17	5.67	0	0	17	7	1	3.22	1.70	0.09
50.4.3.0	0.24	6.00	4.66	0	0	16	6	1	4.22	5.86	0.09
100.4.3.0	0.98	2.67	1.33	0	0	29	2	1	22.38	134.62	11.50
120.4.3.0	1.66	1.89	1.28	0	0	37	4	1	35.35	238.81	47.33
140.4.3.0	0.97	1.62	1.14	0	0	17	2	1	46.38	519.95	24.22
karate	0.18	6.49	2.11	0	0	16	6	1	1.98	1.00	0.07
dolphins	0.32	2.46	0.44	0	0	24	2	1	6.43	15.61	0.16
polbooks	1.31	2.06	0.78	0	0	13	2	1	24.23	172.67	6.66
football	1.49	32.81	24.86	0	0	15	4	1	30.14	237.51	40.90

Table 4. Results of the general set for two fences with radius 1 and 2.

	IDBC								D-ICOP		
Instance	*Time*	*OBJ*	*HeurOBJ*	*UC*	*TUC*	*Cov*	*#M*	*#Re_MIPs*	*IPS*	*PrePro*	*Formu*
20.4.3.0	0.17	259.41	178.02	0	0	17	6	1	0.50	0.09	0.06
30.4.3.0	0.26	247.17	147.00	0	0	29	8	1	1.46	0.65	0.07
40.4.3.0	0.81	113.78	76.87	0	0	32	6	1	3.22	1.70	0.17
50.4.3.0	0.88	95.24	75.59	1	2	40	2	1	4.22	5.86	0.40
100.4.3.0	125.91	35.43	29.14	1	2	73	3	1	22.38	134.62	201.53
120.4.3.0	286.92	25.70	21.80	3	7	84	4	1	35.35	238.81	716.55
140.4.3.0	382.71	21.65	19.01	8	12	98	7	1	46.38	519.95	1762.93
karate	0.85	129.06	70.35	4	4	27	7	1	1.98	1.00	0.10
dolphins	2.49	58.79	47.32	2	2	46	2	1	6.43	15.61	0.83
polbooks	122.51	40.57	33.67	1	2	80	2	1	24.23	172.67	145.29
football	181.98	374.04	358.00	0	1	86	7	1	30.14	237.51	400.04

Table 5. Results of the average set for two fences with radius 1 and 2.

	IDBC								D-ICOP		
Instance	*Time*	*OBJ*	*HeurOBJ*	*UC*	*TUC*	*Cov*	*#M*	*#Re_MIPs*	*IPS*	*PrePro*	*Formu*
20.4.3.0	0.17	181.32	130.33	0	1	17	5	1	0.50	0.09	0.05
30.4.3.0	0.21	193.63	107.44	0	1	29	6	1	1.46	0.65	0.07
40.4.3.0	0.77	80.35	56.82	0	3	33	4	1	3.22	1.70	0.25
50.4.3.0	1.17	64.71	51.99	0	2	41	4	1	4.22	5.86	0.51
100.4.3.0	45.35	25.93	21.74	1	12	74	4	1	22.38	134.62	186.01
120.4.3.0	251.12	18.85	16.18	4	17	84	2	1	35.35	238.81	727.70
140.4.3.0	272.46	16.33	14.28	7	20	99	6	1	46.38	519.95	1514.76
karate	0.83	95.34	68.60	1	2	29	7	1	1.98	1.00	0.16
dolphins	5.84	42.33	36.72	4	8	48	6	1	6.43	15.61	8.87
polbooks	92.27	26.52	22.89	0	6	80	1	1	24.23	172.67	168.08
football	111.87	233.29	225.89	1	7	86	6	1	30.14	237.51	405.92

represents the number of resolved MIPs after adding the violated "big-M" constraints, i.e., the number of iterations in Algorithm 1. For the D-ICOP, the time required to calculate the IPS for each table, the preprocessing time to reduce the size of the IPS, and the time needed to solve the formulation using a DBC algorithm are reported. These times are represented as "IPS", "PrePro", and "Formu", respectively. Note that for tables with the same coverage radius, the "IPS" and "PrePro" values remain constant, as the IPS is not changed while the parameter configuration of the network influence is altered.

The results show that the External Factor Set is significantly easier to solve than the other sets. Particularly, no additional state constraints are added for any of these cases, which indicate that the initial set of states T_1 is sufficient to obtain the minimum absolute eigenvalue of the generator. Moreover, in most cases, it just takes a fraction of a second to find the optimal solution. The algorithm generally adds state constraints for the General and Average parameter sets, although far fewer than the worst-case scenario of 2^n. The Average case is also slightly more challenging to solve, which suggests that having a diverse range of parameters across the network leads to faster solution times.

Note that across all parameter sets the number of repeated MIPs after adding violated "big-M" constraints is mostly one and for a few instances it is zero, which states that just a small portion of the reported time is due to resolving the master problem from scratch at each iteration. Also, the number of violated "big-M" constraints is significantly smaller than $n|F|$, which evidences the effectiveness of rectangular inequalities of Section 4.2. Furthermore, as expected the General and the Average sets of experiments have significantly better objective values in comparison with the External set.

The results of the Heuristic suggest that its effectiveness varies depending on the setting in which network influence can be modified. The value of "OBJ" is found to be 61% greater than "HeurOBJ" on average for the External set, whereas it is 41% and 30% for the General and Average sets, respectively. These results highlight that the Heuristic is more effective in settings where network influence can be modified. Additionally, the results show that the External set covers fewer nodes on average compared with the General and Average sets. This seemingly counterintuitive behavior can be explained by the model's strategy to target a few important nodes in the network rather than several less-connected nodes in the External set. In the other sets, it is not necessary to directly cover important nodes as their influence can be exploited by increasing their influence rates. Therefore, covering more "less-important" nodes and increasing the influence rates of important nodes might be a more optimal strategy in those settings.

When comparing the performance of the IDBC and D-ICOP using Tables 3 to 5, it can be seen that considering only the "Formu" time makes the D-ICOP appear to perform better in instances with fewer than 100 nodes. However, when the "IPS" and "PrePro" are also taken into account, the opposite is true and the IDBC outperforms the D-ICOP. Additionally, even when excluding the "IPS" and "PrePro" the IDBC still performs better than the D-ICOP in larger instances.

We close this section by noting that in Appendix VI, we show results for other combinations of radii, three fences, and the multiple segment PWT formulation, as well as the static coverage mode for two fences with different radii under the General set. These results generally show the same trends as previously discussed, with the only notable difference being that problems become more difficult to solve as there are two or a greater number of fences with equal radii (likely due to symmetry in the MIP formulations). Additionally, it is worth mentioning that under both the General and Average sets, when there are two fences with equal radii, the D-ICOP outperforms the IDBC in terms of the "Formu" time.

8. Conclusions

We introduced the ICOP, an optimization problem for modeling and analyzing network IM problems when the external influence depends on the physical location of both the nodes and ads or Geo-fences. The goal of the decision-maker is to determine the optimal location of a fixed number of ads or Geo-fences, while also adjusting network influence rates within a budget constraint, in order to minimize the time required for the entire network to be influenced. We established that the ICOP is *NP*-hard and developed MIP formulations for three different coverage scenarios. We also devised an IDBC algorithm which utilizes enhancements in the MIP formulations such as sets of valid cuts, reformulated non-linear "big-M" inequalities, and a heuristic to generate a high-quality feasible solution. The concept of critical intersection points from facility location

problems was used to derive a discrete formulation as an alternative approach to evaluate the IDBC performance.

The performance of the IDBC was evaluated through experiments on several instances of social networks with more than a hundred nodes and thousands of arcs, using real location coordinates that were randomly generated in the downtown area of Dallas, Texas. The results of these experiments showed that the IDBC is able to produce optimal and stable solutions for instances within seconds or minutes for the majority of cases. Additionally, we demonstrated the effectiveness of the formulation enhancements by observing that the number of cuts is significantly smaller than 2^n, the number of violated "big-M" constraints is also significantly smaller than $n|F|$ and the number of repeated MIPs is mostly one and sometimes zero. We also presented how significantly each enhancement in the IDBC improves the algorithm's effectiveness. Ultimately, through supplementary experiments, we have established that the ICOP outperforms other models that only consider coverage, as it takes into account the influence of the underlying social network.

Funding

This research is partially funded by the National Science Foundation (NSF) (Award ENG/CMMI # 2145553), by the Air Force Office of Scientific Research (AFORS) (Award # FA9550-22-1-0236), and the Office of Naval Research (ONR) (Award # N00014-19-1-2329).

Notes on contributors

Majid Akhgar received a BS degree in industrial engineering from the Bu-Ali Sina University, Iran, and a PhD degree in industrial engineering from Oklahoma State University. His research interests include network optimization, models for the spread of influence in networks, stochastic modeling, machine learning and supply chain network design problems.

Juan S. Borrero received a BS degree in math and the MS degree in industrial engineering from the Universidad de los Andes in Bogota, Colombia, and a PhD degree in industrial engineering from the University of Pittsburgh. He is an associate professor with the School of Industrial Engineering & Management, Oklahoma State University. His research interests include decision making under uncertainty, bilevel optimization, network optimization, and operations research applied to defense and disaster response.

References

Ackerman, E., Ben-Zwi, O. and Wolfovitz, G. (2010) Combinatorial model and bounds for target set selection. *Theoretical Computer Science*, **411**(44–46), 4017–4022.

Andretta, M. and Birgin, E.G. (2013) Deterministic and stochastic global optimization techniques for planar covering with ellipses problems. *European Journal of Operational Research*, **224**(1), 23–40.

Arief, R.R., Renaldi, F., Umbara, F.R. and Bon, A.T. (2020) Dynamic geofencing in supervision of seller performance, in *Proceedings of the International Conference on Industrial Engineering and Operations Management*, IEOM Society International, Detroit, MI, USA, pp. 1388–1395.

Atamtürk, A. and Narayanan, V. (2010) Conic mixed-integer rounding cuts. *Mathematical Programming*, **122**(1), 1–20.

Bansal, M. and Kianfar, K. (2017) Planar maximum coverage location problem with partial coverage and rectangular demand and service zones. *INFORMS Journal on Computing*, **29**(1), 152–169.

Berman, B. (2016) Planning and implementing effective mobile marketing programs. *Business Horizons*, **59**(4), 431–439.

Borge-Holthoefer, J., Banos, R.A., González-Bailón, S. and Moreno, Y. (2013) Cascading behaviour in complex socio-technical networks. *Journal of Complex Networks*, **1**(1), 3–24.

Borrero, J.S., Akhgar, M. and Krokhmal, P.A. (2021) A scalable Markov chain framework for influence maximization in arbitrary networks. *IEEE Transactions on Network Science and Engineering*, **8**(3), 2372–2387.

Borrero, J.S., Prokopyev, O.A. and Krokhmal, P. (2018) Optimization of cascading processes in arbitrary networks with stochastic interactions. *IEEE Transactions on Network Science and Engineering*, **6**(4), 773–787.

Canbolat, M.S. and von Massow, M. (2009) Planar maximal covering with ellipses. *Computers & Industrial Engineering*, **57**(1), 201–208.

Chen, N. (2009) On the approximability of influence in social networks. *SIAM Journal on Discrete Mathematics*, **23**(3), 1400–1415.

Chen, W., Lakshmanan, L.V. and Castillo, C. (2013) Information and influence propagation in social networks. *Synthesis Lectures on Data Management*, **5**(4), 1–177.

Church, R.L. (1984) The planar maximal covering location problem.(symposium on location problems: in memory of Leon Cooper). *Journal of Regional Science Philadelphia*, **24**(2), 185–201.

Dezső, B., Jüttner, A. and Kovács, P. (2011) Lemon–an open source c++ graph template library. *Electronic Notes in Theoretical Computer Science*, **264**(5), 23–45.

Domingos, P. (2005) Mining social networks for viral marketing. *IEEE Intelligent Systems*, **20**(1), 80–82.

Domingos, P. and Richardson, M. (2001) Mining the network value of customers, in *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, New York, NY, USA, pp. 57–66.

Farahani, R.Z., Asgari, N., Heidari, N., Hosseininia, M. and Goh, M. (2012) Covering problems in facility location: A review. *Computers & Industrial Engineering*, **62**(1), 368–407.

Fischetti, M., Kahr, M., Leitner, M., Monaci, M. and Ruthmair, M. (2018) Least cost influence propagation in (social) networks. *Mathematical Programming*, **170**(1), 293–325.

Fowler, R.J., Paterson, M.S. and Tanimoto, S.L. (1981) Optimal packing and covering in the plane are np-complete. *Information Processing Letters*, **12**(3), 133–137.

García, S. and Marín, A. (2015) Covering location problems, in G. Laporte, S. Nickel, F. S. da Gama (eds.), *Location Science*, first edition, Springer International Publishing, Switzerland, pp. 93–114.

Güney, E. (2019) An efficient linear programming based method for the influence maximization problem in social networks. *Information Sciences*, **503**, 589–605.

Güney, E., Leitner, M., Ruthmair, M. and Sinnl, M. (2021) Large-scale influence maximization via maximal covering location. *European Journal of Operational Research*, **289**(1), 144–164.

Günneç, D., Raghavan, S. and Zhang, R. (2019) Least-cost influence maximization on social networks. *INFORMS Journal on Computing*, **32**(2), 289–302.

Günneç, D., Raghavan, S. and Zhang, R. (2020) A branch-and-cut approach for the least cost influence problem on social networks. *Networks*, **76**(1), 84–105.

Gurobi Optimization. (2021) Gurobi Optimizer Reference Manual. Available at https://www.gurobi.com/documentation/current/refman/index.html.

Hao, J. and Orlin, J.B. (1992) A faster algorithm for finding the minimum cut in a graph, in *Proceedings of the Third Annual ACM-SIAM symposium on Discrete Algorithms*, SIAM, Philadelphia, PA, pp. 165–174.

Hemmati, M., Smith, J.C. and Thai, M.T. (2014) A cutting-plane algorithm for solving a weighted influence interdiction problem. *Computational Optimization and Applications*, **57**(1), 71–104.

Hethcote, H.W. (2000) The mathematics of infectious diseases. *SIAM Review*, **42**(4), 599–653.

Ho, Y.-J., Dewan, S. and Ho, Y.-C. (2020) Distance and local competition in mobile geofencing. *Information Systems Research*, **31**(4), 1421–1442.

IBM ILOG (2021) ILOG Cplex Optimization Studio User's manual. https://www.ibm.com/docs/en/icos/12.9.0?topic=cplex-users-manual, accessed April 2022.

Kempe, D., Kleinberg, J. and Tardos, É. (2003) Maximizing the spread of influence through a social network, in *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, New York, NY, USA, pp. 137–146.

Kempe, D., Kleinberg, J. and Tardos, É. (2015) Maximizing the spread of influence through a social network. *Theory of Computing*, **11**(4), 105–147.

Kitsak, M., Gallos, L.K., Havlin, S., Liljeros, F., Muchnik, L., Stanley, H.E. and Makse, H.A. (2010) Identification of influential spreaders in complex networks. *Nature Physics*, **6**(11), 888–893.

Kulkarni, V.G. (2016) *Modeling and Analysis of Stochastic Systems*. Chapman and Hall/CRC, New York, NY, USA.

Küpper, A., Bareth, U. and Freese, B. (2011) Geofencing and background tracking–the next features in lbss, in *Proceedings of the 41th Annual Conference of the Gesellschaft für Informatik eV*, Gesellschaft für Informatik e.V., Berlin, p. 67.

Leskovec, J. and Krevl, A. (2014, jun) SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data.

Li, D., Li, H., Wang, M. and Cui, J. (2019) k-collective influential facility placement over moving object, in *2019 20th IEEE International Conference on Mobile Data Management (MDM)*, IEEE Press, Piscataway, NJ, pp. 191–200.

Li, G., Chen, S., Feng, J., Tan, K.-l. and Li, W.-s. (2014) Efficient location-aware influence maximization, in *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, ACM, New York, NY, USA, pp. 87–98.

Likas, A., Vlassis, N. and Verbeek, J.J. (2003) The global k-means clustering algorithm. *Pattern Recognition*, **36**(2), 451–461.

Murray, A.T. (2016) Maximal coverage location problem: Impacts, significance, and evolution. *International Regional Science Review*, **39**(1), 5–27.

Murray, A.T. and Tong, D. (2007) Coverage optimization in continuous space facility siting. *International Journal of Geographical Information Science*, **21**(7), 757–776.

Nowzari, C., Preciado, V.M. and Pappas, G.J. (2016) Analysis and control of epidemics: A survey of spreading processes on complex networks. *IEEE Control Systems Magazine*, **36**(1), 26–46.

Pastor-Satorras, R. and Vespignani, A. (2001) Epidemic spreading in scale-free networks. *Physical Review Letters*, **86**(14), 3200.

Raghavan, S. and Zhang, R. (2019) A branch-and-cut approach for the weighted target set selection problem on social networks. *Informs Journal on Optimization*, **1**(4), 304–322.

Raghavan, S. and Zhang, R. (2021a) Weighted target set selection on trees and cycles. *Networks*, **77**(4), 587–609.

Raghavan, S. and Zhang, R. (2021b) Influence maximization with latency requirements on social networks. *INFORMS Journal on Computing*, **34**(2), 710–728.

Raghavan, S. and Zhang, R. (2022) Rapid influence maximization on social networks: The positive influence dominating set problem. *INFORMS Journal on Computing*, **34**(3), 1345–1365.

Rodriguez Garzon, S. and Deva, B. (2014) Geofencing 2.0: Taking location-based notifications to the next level, in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 921–932.

Shaked, M. and Shanthikumar, J.G. (2007) *Stochastic Orders*. Springer Science & Business Media.

Su, S., Li, X., Cheng, X. and Sun, C. (2018) Location-aware targeted influence maximization in social networks. *Journal of the Association for Information Science and Technology*, **69**(2), 229–241.

Taninmiş, K., Aras, N. and Altinel, İ.K. (2022) Improved x-space algorithm for min-max bilevel problems with an application to misinformation spread in social networks. *European Journal of Operational Research*, **297**(1), 40–52.

Taninmiş, K., Aras, N., Altinel, İ.K. and Güney, E. (2020) Minimizing the misinformation spread in social networks. *IISE Transactions*, **52**(8), 850–863.

Tedeschi, D. and Andretta, M. (2021) New exact algorithms for planar maximum covering location by ellipses problems. *European Journal of Operational Research*, **291**(1), 114–127.

Wang, X., Zhang, Y., Zhang, W. and Lin, X. (2016) Efficient distance-aware influence maximization in geo-social networks. *IEEE Transactions on Knowledge and Data Engineering*, **29**(3), 599–612.

Watts, D.J. and Strogatz, S.H. (1998) Collective dynamics of 'small-world' networks. *Nature*, **393**(6684), 440.

Wu, H.-H. and Küçükyavuz, S. (2018) A two-stage stochastic programming approach for influence maximization in social networks. *Computational Optimization and Applications*, **69**(3), 563–595.

Zhang, P., Bao, Z., Li, Y., Li, G., Zhang, Y. and Peng, Z. (2018) Trajectory-driven influential billboard placement, in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ACM, New York, NY, USA, pp. 2748–2757.