VU Research Portal

Large-scale influence maximization via maximal covering location

Güney, Evren; Leitner, Markus; Ruthmair, Mario; Sinnl, Markus

published in

European Journal of Operational Research
2021

DOI (link to publisher)

[10.1016/j.ejor.2020.06.028](https://doi.org/10.1016/j.ejor.2020.06.028)

document version

Publisher's PDF, also known as Version of record

document license

Article 25fa Dutch Copyright Act

[Link to publication in VU Research Portal](#)

citation for published version (APA)

Güney, E., Leitner, M., Ruthmair, M., & Sinnl, M. (2021). Large-scale influence maximization via maximal covering location. *European Journal of Operational Research*, 289(1), 144-164.
<https://doi.org/10.1016/j.ejor.2020.06.028>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

E-mail address:

vuresearchportal.ub@vu.nl

Stochastics and Statistics

Large-scale influence maximization via maximal covering location[☆]Evren Güney^a, Markus Leitner^b, Mario Ruthmair^{c,*}, Markus Sinnl^d^a MEF University, Department of Industrial Engineering, Turkey^b Vrije Universiteit Amsterdam, Department of Supply Chain Analytics, Amsterdam, the Netherlands^c University of Vienna, Department of Statistics and Operations Research, Vienna, Austria^d Johannes Kepler University Linz, Institute of Production and Logistics Management, Linz, Austria

ARTICLE INFO

Article history:

Received 17 May 2019

Accepted 18 June 2020

Available online 24 June 2020

Keywords:

Large scale optimization

Networks

Stochastic programming

Integer programming

Influence maximization

ABSTRACT

Influence maximization aims at identifying a limited set of key individuals in a (social) network which spreads information based on some propagation model and maximizes the number of individuals reached. We show that influence maximization based on the probabilistic independent cascade model can be modeled as a stochastic maximal covering location problem. A reformulation based on Benders decomposition is proposed and a relation between obtained Benders optimality cuts and submodular cuts for correspondingly defined subsets is established. We introduce preprocessing tests, which allow us to remove variables from the model and develop efficient algorithms for the separation of Benders cuts. Both aspects are shown to be crucial ingredients of the developed branch-and-cut algorithm since real-life social network instances may be very large. In a computational study, the considered variants of this branch-and-cut algorithm outperform the state-of-the-art approach for influence maximization by orders of magnitude.

© 2020 Elsevier B.V. All rights reserved.

1. Introduction

Recently, one can observe an increased interest in optimization problems related to the propagation of information in (social) networks, see, e.g., the seminal work in Kempe, Kleinberg, and Tardos (2003) (and its journal version Kempe, Kleinberg, & Tardos (2015)) and an extensive introduction and overview on the topic in Chen, Lakshmanan, and Castillo (2013). Variants of the so-called influence maximization problem typically aim at finding a small set of key individuals in the network which is denoted as “seed set”. External means (e.g., monetary incentives) are used to convince all members of this seed set to spread some information to all directly connected individuals (e.g., followers or friends). This initial spread of information triggers a propagation process in which every active individual forwards (the previously received) information to its neighbors. Thereby, each individual exerts a specific influence to each of its neighbors and a higher influence increases the probability that the receiving individual is activated (e.g., convinced of an information or product). The goal of influence maximization

problems is to choose a seed set of limited size such that the (expected) number of finally active individuals is maximized. Closely related problem variants want to minimize the costs for a seed set given that a certain fraction of all individuals is reached, see, e.g., Fischetti, Kahr, Leitner, Monaci, and Ruthmair (2018). Besides (viral) marketing in (social) networks, similar propagation models have applications in, e.g., epidemiology (Dreyer & Roberts, 2009; Lee, Hasenbein, & Morton, 2015) and analysis of protein-interaction networks (Gibbs & Shmulevich, 2017; Jo, Jung, Moon, & Kim, 2016).

Two main classes of information propagation models have been considered in the literature: (i) threshold models and (ii) cascade models. Both models make use of a given, predefined network whose nodes correspond to network individuals and whose connections to relations between them. In the deterministic linear threshold model proposed by Granovetter (1978), each individual is assigned a hurdle or threshold value and each directed connection between two individuals has a predefined influence value. An individual not contained in the seed set is activated if the (weighted) sum of influence values from active neighbors is at least as high as its hurdle. In the general threshold model arbitrary functions of the influence values from active neighbors and incentive values can be used, see, e.g., Fischetti et al. (2018); Mossel and Roch (2010). In contrast, stochastic cascade models Kempe et al. (2003) assign probabilities to individual connections that represent the chance that an activation attempt of the source individual of a connection succeeds in activating the target individual. Each individual has a

[☆] This work was funded by the Vienna Science and Technology Fund through project ICT15-014 and by the Austrian Research Fund (FWF, Project P26755-N19 and P31366-NBL).

* Corresponding author.

E-mail addresses: guneye@mef.edu.tr (E. Güney), m.leitner@vu.nl (M. Leitner), mario.ruthmair@univie.ac.at (M. Ruthmair), markus.sinnl@jku.at (M. Sinnl).

single attempt to activate its neighbors and these activation attempts are made immediately it gets active. Activation probabilities may change over time and depend on the set of individuals who already tried to influence a particular target individual. The special case of the *independent cascade model* considered in this article is characterized by the fact that activation probabilities do not change over time and are independent of previous attempts to influence a target individual. This special case has received a lot of attention in the literature. To this end we refer, e.g., to [Bharathi, Kempe, and Salek \(2007\)](#) for a competitive two-player variant and [Li, Fan, Wang, and Tan \(2018\)](#) for a survey on fast and scalable heuristics. In [Kempe et al. \(2003\)](#) the authors have shown that the most general variants of the two classes—the general threshold and the general cascade model—are in fact equivalent, whereas this is not the case for the linear threshold and the independent cascade model. There is also a large body of research on the spread of epidemics on random graphs, e.g., based on the Watts-Strogatz model ([Watts & Strogatz, 1998](#)) or the Barabási-Albert model ([Barabási & Albert, 1999](#)). These articles investigate propagation models such as the SIS ([Pastor-Satorras & Vespignani, 2001](#)) or the SIR ([Newman, 2002](#)) model. The main difference to the propagation models considered in our work is that individuals recover and get susceptible (inactive) again (SIS) after being infected (activated) or either recover and get immune or die, which is equivalent to being removed from the network (SIR).

The *influence maximization problem (IMP)* considered in this article is defined on a directed graph $G = (V, A)$ in which nodes V represent individuals and arcs $(i, j) \in A$ correspond to individual i influencing (or activating) j with probability p_{ij} , $0 < p_{ij} \leq 1$. Information propagation is based on the independent cascade model and we assume that active individuals cannot be deactivated again. A feasible solution to the IMP is defined by a seed set $S \subseteq V$ whose size is limited by parameter $K \leq |V|$. We aim to find a seed set S^* which maximizes the expected number of active nodes after the propagation process. As evaluating the expected number of reached nodes for a given seed set is #P-complete (see, e.g., [Kempe et al., 2015](#)), in [Kempe et al. \(2003\)](#) the probabilistic propagation process is approximated by a restricted set Ω of scenarios obtained by Monte-Carlo sampling. Each scenario $\omega \in \Omega$ is represented by *live-arc graph* $G^\omega = (V, A^\omega)$, $A^\omega \subseteq A$, where arc $(i, j) \in A^\omega$ indicates that a previously inactive node j is activated by i in scenario ω if i is active. Arcs $(i, j) \in A$ are included with probability p_{ij} (decided by biased coin flipping) in each scenario, i.e., in arc set A^ω . Let $\Phi^\omega(S)$ denote the number of activated nodes in scenario $\omega \in \Omega$ for a given seed set S , and p_ω be the probability of scenario $\omega \in \Omega$. Then, the variant of the IMP usually considered in the literature (and also in this article) is formally defined as

$$S^* := \operatorname{argmax}_{S \subseteq V, |S| \leq K} \sum_{\omega \in \Omega} p_\omega \Phi^\omega(S).$$

We do not make any assumptions on the sparsity of graph G (and thus of live-arc graphs G^ω , $\forall \omega \in \Omega$). Usually, the (social) networks under consideration are, however, far from being complete and exhibit a rather low average node degree, see [Section 4](#). The live-arc graph G^ω for some scenario $\omega \in \Omega$ can be considered as an inhomogeneous Erdős-Rényi graph ([Erdős & Rényi, 1960](#)) generated from graph G with inhomogeneous arc probabilities (non-existing arcs $(i, j) \in (V \times V) \setminus A$ have probability $p_{ij} = 0$). We also note that [Kempe et al. \(2003\)](#) show that the linear threshold model, in which an individual's hurdle is a random value in $[0, 1]$ and the sum of influence values p_{ij} of arcs (i, j) incoming to j is less or equal to one, can also be represented via live-arc graphs. For each node j at most one incoming arc (i, j) is included in each live arc graph. The probability of including arc (i, j) is equal to p_{ij} , while the probability of including no incoming arc for node j is equal to $1 - \sum_{(i, j) \in A} p_{ij}$. Thus, while we focus on the independent cascade model our ap-

proach can also be used for the linear threshold model after the transformation given above and detailed in [Kempe et al. \(2003\)](#).

1.1. Contribution and outline

In this work, we show how the IMP can be modeled as a *stochastic maximal covering location problem (SMCLP)*. The *maximal covering location problem (MCLP)* ([Church & Velle, 1974](#)) is a classical problem from the area of location science. We present a compact model for the IMP and a reformulation based on Benders decomposition which is the stochastic version of the Benders reformulation for the MCLP recently discussed in [Cordeau, Furini, and Ljubić \(2019\)](#). We also clarify the theoretical relation between the obtained Benders optimality cuts and submodular cuts used for the IMP in the state-of-the-art branch-and-cut algorithm in [Wu and Küçükyavuz \(2018\)](#). Our algorithm is based on a similar set of valid inequalities as a recently proposed Benders decomposition approach for the MCLP ([Cordeau et al., 2019](#)). It turns out, however, that the standard separation approach for the resulting Benders cuts as used in [Cordeau et al. \(2019\)](#) is too time-consuming in our large-scale stochastic setting, and we therefore discuss several speed-up techniques for the separation in our setting. Moreover, we introduce preprocessing tests which allow to remove variables from the model. Both of these ingredients turn out to be crucial for successfully solving very large SMCLP instances resulting from real-life social network instances. Obtained computational results also show that our algorithm outperforms the state-of-the-art in many cases by orders of magnitude.

The paper is organized as follows: In the remainder of this section, we mention previous and related work. [Section 2](#) discusses the relation between influence maximization and stochastic maximal covering location and proposes a cover formulation together with a computationally appealing reformulation. [Section 3](#) describes the algorithm to solve the reformulation, including a description of our new preprocessing procedures. [Section 4](#) shows computational experiments. Finally, [Section 5](#) concludes the paper.

1.2. Previous work

In their seminal paper, [Kempe et al. \(2003\)](#) show that for one particular scenario $\omega \in \Omega$ the real-valued set function $\Phi^\omega(S)$ is non-decreasing and submodular for all $S \subseteq V$. Let $\rho_i^\omega(S) := \Phi^\omega(S \cup \{i\}) - \Phi^\omega(S)$, $\forall S \subseteq V, i \in V$, be the marginal gain achieved by adding element i to set S . According to the general result in [Nemhauser, Wolsey, and Fisher \(1978\)](#), a function $\Phi^\omega(S)$ is non-decreasing and submodular iff $\Phi^\omega(T) \leq \Phi^\omega(S) + \sum_{i \in T \setminus S} \rho_i^\omega(S)$, $\forall S, T \subseteq V$. Furthermore, if there is a cardinality constraint on S , e.g., $|S| \leq K$ (as in our case), a simple greedy heuristic, which starts with $S = \emptyset$ and iteratively adds node $i \notin S$ with maximal marginal gain, gives an $1 - 1/e$ approximation algorithm ([Nemhauser et al., 1978](#)), with e being the base of the natural logarithm. As the objective function of the IMP is a convex combination of submodular functions (which remains submodular), applying the generic greedy algorithm in [Nemhauser et al. \(1978\)](#) to the IMP gives the same approximation ratio. The computation of the expected marginal gain for all nodes can, however, be computationally time-consuming, especially for large graphs and a high number of scenarios. Therefore, several acceleration techniques have been proposed to improve the running time of the greedy algorithm, see, e.g., [Li et al. \(2018\)](#) for a survey on this topic.

The exact solution of the IMP received much less attention: [Güney \(2017\)](#) uses a time-indexed mixed integer linear programming (MILP) formulation for the IMP to model the propagation process in the scenario-specific live-arc graphs. The same author proposes an approximation algorithm based on the linear programming (LP) relaxation of a covering model for the IMP in [Güney](#)

(2019). A similar covering model and an alternative MILP formulation based on cycle elimination have been described in Li, Smith, Dinh, and Thai (2019) which are exact algorithms but lead to long running times for large-sized instances. In the same work a fast approximation algorithm for a generalized variant of the IMP is introduced. In Wu and Küçükyavuz (2018), a MILP formulation is proposed, which is based on the fact that maximizing a non-decreasing and submodular function can be reformulated by using an exponentially-sized family of cutting planes, see Nemhauser and Wolsey (1981) and Nemhauser et al. (1978). To the best of our knowledge, the approach in Wu and Küçükyavuz (2018) is state-of-the-art for solving the IMP and thus will be compared with our methods theoretically and in experiments.

Benders decomposition (BD) methods for different variants of facility location problems have been proposed since the 1980's, see, e.g., Cornuéjols, Nemhauser, and Wolsey (1983), Nemhauser and Wolsey (1981) and Wentges (1996). Computational studies on the efficient use of BD for facility location problems can be found, e.g., in Filippi, Guastaroba, and Speranza (2019), Fischetti, Ljubić, and Sinnl (2016a,b). The survey in Rahmaniani, Crainic, Gendreau, and Rei (2017) shows the recent success of BD for various problems and discusses the ingredients of modern implementations. The literature on BD based solution methods for MCLP variants is, however, rather scarce. Vatsa and Jayaswal (2016) introduced a multi-period MCLP where the number of facilities in each period is uncertain. For a scenario-based uncertainty model, the authors present several MILP formulations and a BD approach for efficiently solving problem instances with up to 500 clients and 20 potential facilities. Their methods and results are, however, not applicable to our setting because of different uncertainty aspects and Benders subproblems, respectively. Additionally and prevalent in many other approaches in the literature, they solve the Benders subproblem only for integer solutions, not for fractional ones to strengthen dual bounds. One reason for this might be the additional computational effort, numerical issues, and convergence problems. The methods discussed in Cordeau et al. (2019) for two complementary deterministic covering problems are most related to our approach, efficiently separating fractional solutions, too. We will refer to the later work repeatedly throughout this paper.

2. Stochastic maximal covering location formulation

In this section, we will establish a close relation between the IMP and a stochastic variant of the maximal covering location problem (Church & Velle, 1974) which has also been observed in Güney (2019); Li et al. (2019). In contrast to the latter works, we not only use this relation to derive an MILP formulation for the IMP but additionally exploit it to obtain a reformulation leading to an effective Benders decomposition algorithm. For this, first observe that the number of activated nodes $\Phi^\omega(S)$ for scenario $\omega \in \Omega$ and seed set $S \subseteq V$ corresponds exactly to the number of nodes which are reachable in live-arc graph G^ω from at least one of the nodes in S . Thus, node i is activated in scenario ω iff set S includes a node from the reachability set $R(\omega, i)$, which is the set of all the nodes from which there is a directed path to i in G^ω . From a different perspective, node j covers node i in scenario $\omega \in \Omega$ if $j \in R(\omega, i)$. The latter point of view reveals the relation to a stochastic maximum covering location problem that is formally shown in Theorem 1.

Theorem 1. *The IMP (in the usually considered discrete-scenario setting) is a special case of a maximal covering location problem with uncertain facility-customer relations.*

Proof. Given sets of facility sites J , demand nodes I with demands $a_i > 0$, $\forall i \in I$, and facility sites $N_i \subset J$ that cover demand node $i \in I$, the MCLP (Church & Velle, 1974) aims to maximize the demand

covered by $p \in \mathbb{N}$ facilities. We observe, that the IMP corresponds to a two-stage stochastic variant of the maximal covering location problem, which aims to maximize the expected demand covered in the second stage while selecting exactly $p = K$ facility locations in the first stage. The sets of facilities and demand nodes are identical in this problem variant and correspond to V (i.e., $I = J = V$) while all demands are equal to one (i.e., $a_i = 1$, $\forall i \in V$). Furthermore, a set of scenarios Ω , each with probability p_ω , $\forall \omega \in \Omega$, is given and the facility sites $N_i^\omega \subseteq V$ covering demand nodes $i \in V$ in scenario $\omega \in \Omega$ are defined by sets $R(\omega, i)$, i.e., some facility j covers customer i iff $j \in R(\omega, i)$. $\square$

For each scenario $\omega \in \Omega$ and node $i \in V$, sets $R(\omega, i)$ can be easily determined by, e.g., a reverse breadth-first search (BFS) starting from i in live-arc graph G^ω . Thus, the result of Theorem 1 can be used to see that formulation (1) is a valid model for the IMP as it is a comparably straightforward scenario-based extension (for the special case described in the proof of Theorem 1) of a classical MILP formulation for the MCLP (Church & Velle, 1974). Thereby, binary variable z_i is equal to one iff node $i \in V$ is part of the seed set, and binary variable x_i^ω is equal to one iff node $i \in V$ is activated in scenario $\omega \in \Omega$.

$$\max \sum_{\omega \in \Omega} p_\omega \sum_{i \in V} x_i^\omega \quad (1a)$$

$$\sum_{i \in V} z_i \leq K \quad (1b)$$

$$x_i^\omega \leq \sum_{j \in R(\omega, i)} z_j \quad \forall i \in V, \forall \omega \in \Omega \quad (1c)$$

$$x_i^\omega \in \{0, 1\} \quad \forall i \in V, \forall \omega \in \Omega \quad (1d)$$

$$z_i \in \{0, 1\} \quad \forall i \in V \quad (1e)$$

Objective function (1a) computes the expected number of activated nodes. The reachability constraint (1c) for node i and scenario ω guarantees that node i can only be active in scenario ω if at least one of the nodes in $R(\omega, i)$ is chosen as seed node.

2.1. Reformulation

Note that depending on the number of scenarios the set of variables x_i^ω , $\forall i \in V, \forall \omega \in \Omega$, and constraints (1c) might be quite large. To effectively solve also large instances, we will now introduce a reformulation of (1) based on Benders decomposition, see, e.g., Cordeau et al. (2019) for a similar approach in the case of the MCLP. First observe, that for fixed integer $\bar{z}_i \in \{0, 1\}$, $\forall i \in V$, the integrality of variables x_i^ω can be relaxed. Moreover, the remaining problem decomposes into one subproblem for each scenario $\omega \in \Omega$. These characteristics make the problem attractive for Benders decomposition. Let μ_ω denote the contribution of scenario $\omega \in \Omega$ to the objective function. Then, we reformulate (1) in the following way:

$$\max \sum_{\omega \in \Omega} p_\omega \mu_\omega \quad (2a)$$

$$\sum_{i \in V} z_i \leq K \quad (2b)$$

$$\mu_\omega \leq \Theta^\omega(\mathbf{z}) \quad \forall \omega \in \Omega \quad (2c)$$

$$z_i \in \{0, 1\} \quad \forall i \in V. \quad (2d)$$

Note that each seed set defined by variables z_i , $\forall i \in V$, which satisfies cardinality constraints (2b) leads to a feasible solution. Thus, no Benders feasibility cuts are needed for the validity of reformulation (2). Inequalities (2c) bound the objective by function $\Theta^\omega(\mathbf{z})$ which gives the maximal number of active nodes in scenario $\omega \in \Omega$ for any solution $\mathbf{z} \in [0, 1]^{|V|}$. Benders optimality cuts to model this function are derived by considering the dual problem of the LP relaxation of (1) with optimal objective value $\Theta^\omega(\mathbf{z})$ for a fixed vector $\mathbf{z}$: Let $\alpha_i^\omega, \forall i \in V, \forall \omega \in \Omega$, be the dual variables associated with upper bound constraints $x_i^\omega \leq 1$, and $\beta_i^\omega, \forall i \in V, \forall \omega \in \Omega$, be the dual variables associated with covering constraints (1c). Then, for one particular scenario $\omega \in \Omega$ the dual is

$$\Theta^\omega(\mathbf{z}) := \min \left\{ \sum_{i \in V} \left(\alpha_i^\omega + \beta_i^\omega \sum_{j \in R(\omega, i)} \bar{z}_j \right) : (\alpha^\omega, \beta^\omega) \in P^\omega \right\} \quad (3)$$

with

$$P^\omega := \{(\alpha^\omega, \beta^\omega) : \alpha^\omega + \beta^\omega \geq \mathbf{1}, \alpha^\omega, \beta^\omega \geq \mathbf{0}\}. \quad (4)$$

It is easy to see that there exists an optimal solution to (3) in which $(\alpha_i^\omega, \beta_i^\omega) \in \{(1, 0), (0, 1)\}$ for each $i \in V$, depending on the corresponding coefficients in the objective function. If $\sum_{j \in R(\omega, i)} \bar{z}_j < 1$, then $\beta_i^\omega = 1$ (and $\alpha_i^\omega = 0$), otherwise, $\alpha_i^\omega = 1$ (and $\beta_i^\omega = 0$). Thus, the dual can be solved by inspection for each scenario $\omega \in \Omega$. Note that we always obtain an integer dual optimal solution, also for fractional values of $\mathbf{z}$. Also note that in case $\sum_{j \in R(\omega, i)} \bar{z}_j = 1$ for some node i , we can freely choose whether to set $\alpha_i^\omega = 1$ or $\beta_i^\omega = 1$, leading to different Benders cuts, cf. Cordeau et al. (2019). We decided to set $\alpha_i^\omega = 1$ in such a case. This strategy which leads to sparser cuts is also supported by the computational results for the deterministic MCLP shown in Cordeau et al. (2019).

We denote by $C^\omega(\mathbf{z}) := \sum_{i \in V} \alpha_i^\omega$ the constant and by $c_j^\omega(\mathbf{z}) := \sum_{i: j \in R(\omega, i)} \beta_i^\omega$ the coefficient of variable z_j , $\forall j \in V$, in Benders optimality cuts

$$\mu_\omega \leq C^\omega(\mathbf{z}) + \sum_{j \in V} c_j^\omega(\mathbf{z}) z_j. \quad (5)$$

2.2. Relation to submodular cuts

The decomposition approach in Wu and Küçükyavuz (2018) uses optimality cuts based on the submodularity and monotonicity of function Φ^ω , cf. Section 1.2. Using the variables and notation given above, they are defined by

$$\mu_\omega \leq \Phi^\omega(S) + \sum_{j \in V \setminus S} \rho_j^\omega(S) z_j, \quad \forall S \subseteq V : |S| \leq K, \forall \omega \in \Omega. \quad (6)$$

While only inequalities (6) for sets S of size exactly K are needed for correctness of the formulation, those associated with sets S of smaller size can be computationally beneficial as observed in Nemhauser and Wolsey (1981). In fact, the latter inequalities potentially strengthen the LP relaxation of the formulation including only submodular cuts (6) for sets S of size exactly K , as shown in the following example: Consider graph G with nodes $V = \{1, 2, 3, 4\}$, arcs $A = \{(1, 4), (2, 4), (3, 4)\}$, a maximal seed set size $K = 2$, and influence probability $p_{ij} = 1, \forall (i, j) \in A$, and a single scenario ω with $G^\omega = G$. An optimal integer solution has objective value 3 and can be obtained by selecting any two of the three nodes 1, 2, 3. A formulation including all submodular cuts for sets of size exactly $K = 2$ leads to an optimal LP solution $\bar{z}_1 = \bar{z}_2 = \bar{z}_3 = \frac{2}{3}, \bar{z}_4 = 0$, with objective $\mu = \frac{10}{3}$ (bounded above by the six submodular cuts for all pairs of nodes). Submodular cut (6) for the single node set $S = \{4\}$ is defined as $\mu \leq 1 + z_1 + z_2 + z_3$ and thus is violated by the LP solution above. Adding the latter cut to the model results in an optimal integer LP solution with value $\mu = 3$.

Theorem 2. The Benders optimality cut (5) corresponding to a given (fractional) solution $\mathbf{z}$ is equivalent to the submodular cut (6) for set $S := \{i \in V : \sum_{j \in R(\omega, i)} \bar{z}_j \geq 1 \text{ and } \sum_{j \in R(\omega, k)} \bar{z}_j < 1, \forall k \in R(\omega, i) \setminus \{i\}\}$.

Proof. Note that different sets can lead to the same submodular cut, e.g., $S' := \{i \in V : \sum_{j \in R(\omega, i)} \bar{z}_j \geq 1\} \supseteq S$ results in the same inequality as set S which, however, is a smallest one since it does not contain nodes which are reachable from other nodes in S (due to the second term in the definition of S). In other words, if $i \in S$ and for some node $k \in R(\omega, i) \setminus \{i\}$ the condition $\sum_{j \in R(\omega, k)} \bar{z}_j \geq 1$ would hold, then $k \in S$ and thus i does not need to be in S since it is anyway activated by k .

We show that $C^\omega(\mathbf{z}) = \Phi^\omega(S)$ and $c_j^\omega(\mathbf{z}) = \rho_j^\omega(S), \forall j \in V$. By definition of S , $\sum_{j \in R(\omega, i)} \bar{z}_j \geq 1$ holds for all nodes $i \in V$ reachable from some node in seed set S . In this case, we set $\alpha_i^\omega = 1$ and $\beta_i^\omega = 0$. For all nodes i which cannot be activated by S , $\sum_{j \in R(\omega, i)} \bar{z}_j < 1$ holds and thus, $\alpha_i^\omega = 0$ and $\beta_i^\omega = 1$. Therefore, $C^\omega(\mathbf{z}) = \sum_{i \in V} \alpha_i^\omega$ corresponds to the number of nodes activated by seed set S , i.e., $C^\omega(\mathbf{z}) = \Phi^\omega(S)$. For one particular node $j \in V$, coefficient $c_j^\omega(\mathbf{z}) = \sum_{i: j \in R(\omega, i)} \beta_i^\omega$ denotes the number of not yet active nodes which would be activated by node j if j is added to the seed set which corresponds to the marginal gain of node j , i.e., $c_j^\omega(\mathbf{z}) = \rho_j^\omega(S)$. Note that the marginal gain of nodes from S and all other active nodes is zero. $\square$

Corollary 1. Benders optimality cuts (5) associated to integer solutions $\mathbf{z}$ and submodular cuts (6) for correspondingly defined seed sets $S := \{i \in V : \bar{z}_i = 1\}$ are equivalent.

Even though the two sets of inequalities are equivalent for integer solutions, compared to Wu and Küçükyavuz (2018) our modeling approach allows an efficient exact separation routine for fractional solutions which does not rely on solving the LP relaxation. These Benders cuts based on fractional solutions turn out to improve the computational results significantly, see Section 4.

Interestingly, set S obtained in Theorem 2 can potentially have any size, also larger than K , even for solutions with $\sum_{i \in V} \bar{z}_i = K$. For the LP solution $\bar{z}_1 = \bar{z}_2 = \bar{z}_3 = \frac{2}{3}, \bar{z}_4 = 0$, in the example above, by applying Theorem 2 we obtain $S := \{4\}$ leading exactly to the submodular inequality mentioned above which cuts off this LP solution. To summarize, the Benders cuts for fractional solutions potentially lead to submodular cuts for sets of size different to K , which might not be generated otherwise since integral solutions found by general-purpose heuristics included in the MILP solver or by the LP relaxation in some branch-and-bound node very likely will exploit the budget and will have size K .

3. Solution algorithm

To solve the IMP we use a branch-and-cut algorithm based on the Benders reformulation (2). In general, it is possible to start with a relaxed master problem which does not include any of the Benders cuts (2c). However, it turned out to be computationally advantageous to initially include some cuts to bound variables μ_ω from above, otherwise the first solution of the master problem would be unbounded. Thus, we add the following inequalities which correspond to submodular cuts for $S = \emptyset$, i.e.,

$$\mu_\omega \leq \sum_{j \in V} \Phi^\omega(\{j\}) z_j, \quad \forall \omega \in \Omega. \quad (7)$$

In the remainder of this section we discuss issues related to the reachability sets for each node in each scenario and the

separation of Benders cuts (5) for integral and fractional solutions, respectively.

3.1. Determining reachability sets

Reachability sets $R(\omega, i)$ are essential for solving the Benders subproblem and obtaining violated Benders cuts (5). In principle, the reachability sets $R(\omega, i)$ may be pre-computed and stored for all scenarios $\omega \in \Omega$ and nodes $i \in V$. The required memory of $\mathcal{O}(|V|^2|\Omega|)$ may, however, become too large for real-world social networks with many nodes and when considering a high number of scenarios. Instead of pre-computing and storing sets $R(\omega, i)$ one could alternatively determine them on-the-fly during the separation of Benders cuts (5). The resulting runtime of $\mathcal{O}(|V| + |E|)|V||\Omega|)$ per call of the separation routine might be too time consuming. As a consequence we use an intermediate approach based on external parameter `MemLimPerScen` which denotes the maximally allowed memory consumption per scenario. More precisely, we pre-compute and store reachability sets for a subset of nodes $V^\omega \subseteq V$ in each scenario $\omega \in \Omega$ such that the required memory does not exceed `MemLimPerScen`. Sets $R(\omega, i)$ for remaining nodes $i \in V \setminus V^\omega$ are determined on-the-fly during the cut separation.

For each scenario $\omega \in \Omega$ and node $i \in V$, set $R(\omega, i)$ is determined using reverse BFS. If there is enough memory available for the current scenario, $R(\omega, i)$ is stored for later use and the next node is considered. Otherwise, the current set $R(\omega, i)$ is discarded and the algorithm continues with the next scenario. Two further enhancements are considered in our implementation that help reduce the runtime. The first one is based on the observation that a reverse BFS on G^ω for some node $i \in V$ may encounter a node $j \in V^\omega$ for which $R(\omega, j)$ has already been computed and stored. Since $R(\omega, j) \subseteq R(\omega, i)$ in this case, we add set $R(\omega, j)$ to the (partial) set $R(\omega, i)$ and do not further proceed the reverse BFS from any node in $R(\omega, j)$. The second one is to immediately stop a reverse BFS iteration and discard the (partial) set $R(\omega, i)$ if the memory limit is reached instead of first finishing the computation of this set.

To keep the overall approach as simple and fast as possible set V^ω is determined by simply enumerating through the nodes $i \in V$ in increasing order with respect to their index and computing and storing the associated set $R(\omega, i)$ as long as the memory limit is not reached for the current scenario $\omega \in \Omega$. In Section 4.5 we also test an alternative way of determining V^ω which, however, leads to no improvement.

3.2. Separation of Benders cuts

Algorithm 1 describes our separation routine of Benders cuts (5) when given a current solution $(\bar{z}, \bar{\mu})$ to the LP relaxation of (2). For each scenario $\omega \in \Omega$, we first determine the set $R^\omega(S)$ of all nodes that can be reached from set $S = \{i \in V : \bar{z}_i = 1\}$. We set $(\alpha_i^\omega, \beta_i^\omega) = (1, 0)$, $\forall i \in R^\omega(S)$, since there always exists an optimal solution to (3) with these values, cf. Section 2.1. Next, we evaluate $\sum_{j \in R(\omega, i)} \bar{z}_j$ for all remaining nodes $i \in V \setminus R^\omega(S)$. If this sum is at least one, we set $(\alpha_i^\omega, \beta_i^\omega) = (1, 0)$, and $(\alpha_i^\omega, \beta_i^\omega) = (0, 1)$ otherwise. Note that the second step is simplified if the current LP solution is integral, i.e., if $\bar{z}_i \in \{0, 1\}$ holds for all $i \in V$. In this case $\sum_{j \in R(\omega, i)} \bar{z}_j$ is equal to zero and thus $(\alpha_i^\omega, \beta_i^\omega) = (0, 1)$ for all nodes $i \in V \setminus R^\omega(S)$. Instead of storing the full solution to the dual of (2), **Algorithm 1** uses C^ω and c_j^ω , $\forall j \in V$, to keep track of $\sum_{i \in V} \alpha_i^\omega$ and $\sum_{i: j \in R(\omega, i)} \beta_i^\omega$, respectively. Finally, we stop the reverse BFS for node $j \in V \setminus R^\omega(S)$ as soon as the sum of relevant z-values reaches one (further increasing this sum does not change the overall result). For the sake of readability, these improvements are omitted in **Algorithm 1**.

Algorithm 1: separate.

Input: live-arc graphs $G^\omega = (V, A^\omega)$, V^ω , $\forall \omega \in \Omega$, solution $(\bar{z}, \bar{\mu})$

Output: set $\mathcal{C}$ of Benders cuts (5) violated by $(\bar{z}, \bar{\mu})$

```

1  $S = \{i \in V : \bar{z}_i = 1\};$ 
2  $\mathcal{C} = \emptyset;$ 
3 for  $\omega \in \Omega$  do
4   obtain active node set  $R^\omega(S)$  in  $G^\omega$ ;
5    $C^\omega = |R^\omega(S)|;$ 
6    $c_j^\omega = 0, \forall j \in V;$ 
7   for  $i \notin R^\omega(S)$  do
8     if  $\sum_{j \in R(\omega, i)} \bar{z}_j \geq 1$  then
9        $C^\omega = C^\omega + 1;$ 
10    else
11       $c_j^\omega = c_j^\omega + 1, \forall j \in R(\omega, i);$ 
12    if  $\bar{\mu}_\omega > C^\omega + \sum_{j \in V} c_j^\omega \bar{z}_j$  then
13       $\mathcal{C} = \mathcal{C} \cup \{\mu_\omega \leq C^\omega + \sum_{j \in V} c_j^\omega z_j\};$ 
14 return  $\mathcal{C};$ 

```

3.3. Preprocessing

We propose and apply two preprocessing steps that aim to reduce the size of the model and the density of Benders cuts (5), and as a consequence reduce solution times and memory consumption.

P1 The first step eliminates seed set variables z based on the observation that node $i \in V$ is dominated by node $j \in V \setminus \{i\}$, if $j \in R(\omega, i)$ in all scenarios $\omega \in \Omega$. In this case, it is better (or equivalent) to include j in the seed set instead of i since all nodes reached by i can also be reached by j (in all scenarios). Thus, variable z_i can be eliminated from the model. If i and j dominate each other, we can eliminate any of the two associated variables which also reduces symmetries in the solution space.

P2 Observe that the covering constraint (1c) is binding in any optimal solution if $i \in V$ is a node with no incident arcs in A^ω in some scenario $\omega \in \Omega$, i.e., $x_i^\omega = z_i$. Thus, we remove the covering constraint and replace variable x_i^ω by z_i in the objective function for such nodes. As a consequence, variables α_i^ω and β_i^ω do not exist in dual (3). Since node i does not appear in any set $R(\omega, j)$ for $j \neq i$, the coefficient of variable z_i in any Benders cut (5) for scenario ω is zero. Observe that this preprocessing step completely avoids the need to consider variables associated to nodes without incident arcs in Benders cuts. An alternative way to exploit this structural property has been used in Cordeau et al. (2019) to strengthen the Benders cuts in the context of the MCLP.

4. Computational experiments

Each experiment has been performed on a single core of an Intel Xeon E5-2670v2 machine with 2.5 GHz. The algorithms are implemented in C++ and IBM ILOG CPLEX 12.7 with default settings is used as branch-and-cut framework and LP solver. A time limit of 3 600 seconds and, unless otherwise stated, a memory limit of 30 GB has been set for each test run with parameter `MemLimPerScen` set to $8/|\Omega|$ GB.

4.1. Instances

We test our solution algorithms on the four real-world social networks used in Wu and Küçükyavuz (2018), see Table 1, and process them in the same way to allow a fair comparison: In case the instance graph is undirected, we add two directed arcs (i, j)

Table 1
Real-world social networks.

Instance	$ V $	$ \hat{A} $	$ A $	Description
MSG	1899	59 835	20 296	messaging network of users at UC-Irvine Panzarasa, Opsahl, and Carley (2009)
GNU	10 879	79 988	79 988	snapshot 04 of Gnutella filesharing network Ripeanu and Foster (2002)
HEP	15 233	117 782	62 774	high energy physics paper citation network Chen et al. (2009)
ENRON	36 692	367 662	367 662	e-mail communication network from Enron Leskovec, Lang, Dasgupta, and Mahoney (2009)

Fig. 1. Performance profiles of solution runtimes with stopping criterion of 1% and 0% optimality gap, respectively, for the real-world instances tested in Wu and Küçükyavuz (2018).

and (j, i) for each undirected edge $\{i, j\}$. Each arc $(i, j) \in A$ is assigned the same influence probability $p_{ij} = 0.1$. However, instances MSG and HEP include multiple parallel arcs for some $(i, j) \in A$ which are handled in the following way: When generating a simple live-arc graph G^ω for some scenario $\omega \in \Omega$, for each appearance of $(i, j) \in A$ we decide with probability p_{ij} whether that arc is part of A^ω or not. Alternatively, we may compute a combined probability meaning that at least one of the arcs $(i, j) \in A$ appears in G^ω , i.e., $p'_{ij} = 1 - (1 - p_{ij})^{n_{ij}}$, where n_{ij} corresponds to the number of parallel arcs $(i, j) \in A$. Columns $|\hat{A}|$ and $|A|$ in Table 1 show the total number of arcs (including parallel arcs) and the number of unique arcs for each instance, respectively. Instances MSG, GNU, and ENRON are available for download from the SNAP database¹, while instance HEP can be obtained from the website of one of the authors of Chen, Wang, and Yang (2009)².

Moreover, we generated a set SW of five directed small-world graphs (Watts & Strogatz, 1998) with 1000 nodes and rewiring probability 0.3 for each average node out-degree $\delta^+ \in \{8, 16\}$. Influence probabilities p_{ij} for arcs $(i, j) \in A$ are either set to a common value in $\{0.01, 0.05, 0.1\}$ (similar as above) or are β -distributed with parameters (2,18) (mean value 0.1).

4.2. Algorithmic variants and settings

Based on the branch-and-cut algorithm described in Section 3 we compare three variants differing in the Benders cut separation:

- BEN: In this setting, we separate cuts only for integer solutions.
- BENFR: BEN with additional separation of fractional solutions but only in the root node of the branch-and-bound tree.
- BENF: BENFR with separation of fractional solutions in all branch-and-bound nodes.

In all settings we initially add submodular cuts (7) for empty sets. In both BENFR and BENF, we stop separation of fractional cuts in some branch-and-bound node as soon as the absolute decrease of the dual bound compared to the previous iteration falls below 0.001.

4.3. Comparison with state-of-the-art

In this section, we compare our solution algorithm with the delayed row-generation approach based on submodular cuts (6) in Wu and Küçükyavuz (2018) which to the best of our knowledge is state-of-the-art for solving the IMP. The authors in Wu and Küçükyavuz (2018) iteratively solve their formulation with a subset of submodular cuts to integer optimality, identify and add violated submodular cuts, and update bounds until the global optimality gap is closed. Note, however, that a 1% optimality gap threshold is set for solving the (incomplete) formulation in each iteration. This potentially leads to suboptimal solutions in the end also in cases where the global optimality gap (outer loop) is closed completely. For comparison reasons, we re-implemented their approach to perform tests in which we solve in each iteration the formulation to optimality. Furthermore, the authors in Wu and Küçükyavuz (2018) argue that adding submodular cuts (7) for empty sets a priori to the model does not help to (significantly) improve the running time of their approach or reduce the number of generated cuts. This might be explained by the fact that a formulation without any submodular cuts will most probably lead to an optimal solution $\bar{z} = \mathbf{0}$. Thus, the separation in the first iteration will identify and add exactly the submodular cuts (7) for empty sets. Preliminary test showed that both variants proceed quite similarly (within the variability of modern MILP solvers). To avoid this initial iteration we instead immediately add cuts (7). We denote the approach from Wu and Küçükyavuz (2018) by WK in the following comparison. Tests are performed with the same settings as in Wu and Küçükyavuz (2018), i.e., $K \in \{2, 3, 4, 5\}$ and $|\Omega| \in \{100, 200, 300, 500\}$. Fig. 1 and b give performance profiles comparing the running time of our algorithms with WK for 1% and 0% optimality gap threshold, respectively. The optimality gap is given by $100 \cdot (UB - LB)/LB$, where UB and LB are best upper and lower bound found by a solution algorithm, respectively. Note that for our algorithms optimality gap thresholds > 0 apply to the single branch-and-cut execution. As can be clearly seen in the figures, it seems to be beneficial to integrate the generation of Benders cuts within a cutting plane approach in a single branch-and-cut iteration. Our variant BEN only separating integer solutions performs best here, immediately followed by BENFR adding fractional Benders cuts only in the root node, both outperforming algorithm WK by at least one order of magnitude. Detailed results can be found

¹ <https://snap.stanford.edu/data/>, accessed on 2018-10-18.

² <https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/weic-graphdata.zip>, accessed on 2018-10-18.

Fig. 2. Performance profiles of solution runtimes and optimality gaps on the real-world instances, grouped by target set size K .

in the Appendix in Tables A.4 and A.5. These instances where the target set size K is relatively small and the influence probability is set to a high value of $p = 0.1$ for all arcs turned out to be quite easy to solve. Because of the high influence probability, the live-arc graphs usually contain a giant (weakly) connected component, which is also very similar among the scenarios, and several quite

small or even singleton components. Thus, with a small target set size $K \leq 5$ finding an optimal solution is not too difficult: It seems sufficient to select a promising target node in the giant component to activate most of the other nodes of the giant component and distribute the few residual target nodes to the smaller components improving the objective only slightly.

Fig. 3. Performance profiles of solution runtimes and optimality gaps on the SW instances, grouped by target set size K .

4.4. Extensive performance comparison

Due to the observations in the last section, we increase the difficulty of the real-world instances by considering also smaller influence probabilities $p \in \{0.01, 0.05, 0.1\}$ (in one particular instance all arcs still have the same base probability, resulting in different probabilities in case of parallel arcs) and higher target set sizes $K \in \{2, 5, 10, 15, 25\}$. The number of scenarios is set to $|\Omega| \in \{250, 500, 750\}$. Smallworld graphs SW are tested only for $K \in \{10, 15, 25\}$ but for the same set of influence probabilities with the additional case of β -distributed values to also deal with the (more realistic) situation of varying arc probabilities. In all further experiments we only consider a 0% optimality gap threshold in our algorithms. Figs. 2 and 3 show performance profiles of solution runtimes and optimality gaps grouped by target size K while Figs. 4 and 5 show profiles grouped by influence probability p . Detailed results can be found in the appendix in Tables A.6–A.11. We note that for approach WK, instance ENRON, $p = 0.1$, and $|\Omega| = 750$, the initial model could not be built within the given time limit, as the generation of the initial cuts (7) required too much time without our enhancements, see Section 3.1 (leading to optimality gap ∞ in the performance profiles). In the tables, we also report the primal gaps given by $100 \cdot (LB' - LB) / LB$, where LB' is the best known lower bound (found by any algorithm).

We immediately observe that enlarging the target set size increases the solution runtime and also the remaining optimality gap in case the time limit is reached. This result is in line with the discussion in the last section and is also quite intuitive since a

larger target set size implies a larger set of feasible solutions. With respect to our algorithmic variants, it turns out that when K increases separating fractional solutions gets more and more important, i.e., the performance of BENF/BENFR significantly improves compared to BEN. This behavior can be explained by the fact that the dual bound obtained from the LP relaxation of (2) deteriorates when K increases. This can be clearly observed in the numbers of needed branch-and-bound nodes. Variants BENF and BENFR perform very similar with respect to overall runtimes and remaining optimality gaps. Despite differences in numbers of branch-and-bound nodes and cuts generated, none of them seems to outperform the other on the tested instances and parameters. Overall, the results seem to indicate slight advantages of BENFR over BENF on the real-world instances while BENF seems to perform slightly better on average and in particular for the SW instances. Overall, this seems to indicate that the relevance of cuts added at nodes different from the root node and whether the additional effort to separate them pays off, depends on the considered instance and its parameters. When relating different influence probabilities, interestingly the intermediate case of $p = 0.05$ is more difficult than the other two cases. Mostly, value $p = 0.1$ results in one giant and several very small or singleton components in each scenario (as discussed above) while $p = 0.01$ leads to extremely sparse live-arc graphs containing a large number of singleton components. Both extreme cases are therefore not difficult to solve whereas intermediate values generate many components of different sizes that also differ a lot more between scenarios. This can especially be observed in the extremely large remaining optimality gaps of up to

Fig. 4. Performance profiles of solution runtimes and optimality gaps on the real-world instances, grouped by p .

20% for the SW instances with average degree 16, see Table A.11. To gain insights into the influence of the graph density on the difficulty of instances, we compare the results for the SW instances with average degrees 8 and 16, see Tables A.10 and A.11, respectively. These results reveal an interesting relation between solution times, graph density, and influence probabilities. We observe that the difficulty of instances with average degree 8 is highest when $p = 0.1$, while most of the instances can be solved quite fast by BENF and BENFR for smaller values of p . On the contrary, instances with average degree 16 are very difficult to solve for these two algorithms when $p = 0.05$, but comparably easy if $p \in \{0.01, 0.1\}$. We conclude that the density of an underlying graph does not seem to be a major factor influencing the solution times. Instead, the combination of graph density and influence probability is a decisive criterion that has a strong influence on whether the resulting live-arc graphs are extremely sparse (e.g., sparse underlying graphs with small influence probabilities), typically result in a giant component (e.g., dense graphs with comparably high influence probabilities), or none of these two (e.g., sparse graphs with relatively high influence probabilities or dense graphs with medium influence probabilities). The SW instances with β -distributed probabilities show quite similar results to the $p = 0.1$ case which can be explained by the fact that the mean value of the chosen distribution corresponds exactly to 0.1. Finally, we remark that while our results show that the algorithms' performance is related to connectivity properties of the live-arc graphs, the influence of other aspects such as their node degree distributions is not so clear. Consider the following two extreme live-arc graphs: A complete graph and a directed cycle. The two graphs have very different

node degrees but their reachability sets are the same: Each node is reachable from every other node. Since our algorithms mainly work on those reachability sets, we expect a strong correlation between the runtimes and the structure of the reachability sets.

Increasing the number of scenarios clearly leads to a better approximation of the IMP but also to a much higher time and memory consumption. To summarize, among our algorithmic variants BENF seems to be the most promising one, especially for the SW instances, while BENFR with its more conservative and parsimonious separation strategy seems to be better suited for the real-world instances. Common to both winners, our efficient separation of fractional solutions seems to pay off significantly.

4.5. Influence of (partially) storing reachability sets

As discussed in Section 3.1, reachability sets for each node and each scenario (if stored completely) potentially consume a large amount of memory. In this section we evaluate our strategy of partially storing the reachability sets for (tight) memory limits. With the default setting of reserving 8 GB for all reachability sets for an instance we observe that for most considered real-world and small-world instances and numbers of scenarios the limit is not reached and thus all sets could be stored. Tables 2 and 3 show next to preprocessing results (see Section 4.6) also a column named "stored" with the relative average amounts of stored reachability sets (over all scenarios). For the largest instance ENRON and higher influence probabilities (leading to denser live-arc graphs and potentially larger reachability sets), however, in some cases even less than ten percent of all reachability sets can be stored.

Fig. 5. Performance profiles of solution runtimes and optimality gaps on the SW instances, grouped by influence probability p .

Therefore, we perform additional tests on instance ENRON to compare the solution runtimes of algorithm BENF for different amounts of memory reserved for reachability sets controlled via parameter `MemLimPerScen`: (i) BENF0 with `MemLimPerScen` = 0 (no sets are stored and thus repeatedly determined on-the-fly), (ii) BENF with `MemLimPerScen` = $8/|\Omega|$ (default setting as described above), and (iii) BENF30 with `MemLimPerScen` = $30/|\Omega|$ (30 GB are sufficient to store all sets for this instance). The natural question arises which of the reachability sets should be stored and which should be determined on-the-fly in each scenario in case of tight memory limits. In setting BENF for each scenario we iterate through the set of nodes ordered by increasing node index and store the associated reachability sets until `MemLimPerScen` is reached. To see whether the ordering has an influence on the solution runtime, we alternatively consider setting BENFS sorting the nodes by non-increasing out-degree, thus favoring the storage of potentially large reachability sets. Fig. 6 shows performance profiles comparing all four variants. It is easy to see that storing no

Fig. 6. Performance profiles of solution runtimes for different settings of `MemLimPerScen` and sorting strategies.

sets (BENF0) leads to significantly higher solution runtimes even though determining the sets on-the-fly can be done quite efficiently in linear time. For $p = 0.1$ and 750 scenarios, even building the formulation with initial cuts (7) could not be done within the

Table 2

Preprocessing results for real-world instances with relative amounts of stored reachability sets (“stored”), eliminated seed set variables ($P1$), and eliminated activation variables ($P2$), averaged over all scenarios.

name	$ V $	$ A $	$ \Omega $	p	stored	$P1$	$P2$
MSG	1899	59,835	250	0.01	1.00	0.01	0.63
				0.05	1.00	0.08	0.45
				0.10	1.00	0.16	0.36
			500	0.01	1.00	0.00	0.63
				0.05	1.00	0.08	0.45
				0.10	1.00	0.15	0.35
			750	0.01	1.00	0.00	0.63
				0.05	1.00	0.07	0.45
				0.10	1.00	0.14	0.35
			250	0.01	1.00	0.00	0.77
				0.05	1.00	0.00	0.52
				0.10	1.00	0.00	0.37
GNU	10,879	79,988	500	0.01	1.00	0.00	0.77
				0.05	1.00	0.00	0.52
				0.10	1.00	0.00	0.37
			750	0.01	1.00	0.00	0.77
				0.05	1.00	0.00	0.52
				0.10	1.00	0.00	0.37
			250	0.01	1.00	0.00	0.79
				0.05	1.00	0.00	0.59
				0.10	1.00	0.01	0.43
			500	0.01	1.00	0.00	0.79
				0.05	1.00	0.00	0.59
				0.10	1.00	0.01	0.43
HEP	15,233	117,782	750	0.01	1.00	0.00	0.79
				0.05	1.00	0.00	0.59
				0.10	1.00	0.01	0.43
			250	0.01	1.00	0.00	0.79
				0.05	1.00	0.00	0.59
				0.10	1.00	0.01	0.43
			500	0.01	1.00	0.00	0.79
				0.05	1.00	0.00	0.59
				0.10	0.97	0.01	0.43
			250	0.01	1.00	0.00	0.80
				0.05	0.14	0.01	0.61
				0.10	0.07	0.03	0.46
ENRON	36,692	367,662	500	0.01	1.00	0.00	0.80
				0.05	0.06	0.01	0.61
				0.10	0.04	0.03	0.46
			750	0.01	1.00	0.00	0.80
				0.05	0.04	0.01	0.61
				0.10	0.03	0.02	0.46

Table 3

Preprocessing results for SW instances with relative amounts of stored reachability sets (“stored”), eliminated seed set variables ($P1$), and eliminated activation variables ($P2$), averaged over all scenarios.

avg. δ^+	$ V $	$ A $	$ \Omega $	p	stored	$P1$	$P2$
8	1000	8000	250	0.01	1.00	0.00	0.72
				0.05	1.00	0.00	0.37
				0.10	1.00	0.00	0.16
			500	β	1.00	0.00	0.16
				0.01	1.00	0.00	0.72
				0.05	1.00	0.00	0.37
			750	0.10	1.00	0.00	0.16
				β	1.00	0.00	0.16
				0.01	1.00	0.00	0.72
			250	0.05	1.00	0.00	0.37
				0.10	1.00	0.00	0.16
				β	1.00	0.00	0.16
16	1000	16,000	500	0.01	1.00	0.00	0.52
				0.05	1.00	0.00	0.14
				0.10	1.00	0.00	0.02
			750	β	1.00	0.00	0.02
				0.01	1.00	0.00	0.52
				0.05	1.00	0.00	0.14
			250	0.10	1.00	0.00	0.02
				β	1.00	0.00	0.02
				0.01	1.00	0.00	0.52
			500	0.05	1.00	0.00	0.14
				0.10	1.00	0.00	0.02
				β	1.00	0.00	0.02

time limit when storing no sets (BENF0). Furthermore, reserving huge amounts of memory to store all sets (BENF30) and selecting the sets to store based on some more sophisticated criterion

(BENFS), respectively, show no significant benefits compared with our default strategy of partially storing the sets based on index ordering (BENF).

Fig. 7. Performance profiles of solution runtimes with and without (-NOP) preprocessing.

4.6. Preprocessing results

In this section we briefly discuss the influence of preprocessing on our sets of test instances. Tables 2 and 3 show the average numbers of eliminated node variables over all scenarios for real-world and SW instances, respectively.

The requirements for an elimination of a seed set variable ($P1$) are by definition quite hard to fulfill since the corresponding dominance rule has to hold in all scenarios, i.e., the more scenarios the smaller the chance for an elimination. Also, higher graph density is favorable for this preprocessing step which is clearly visible for instance MSG in Table 2 in which already the base graph is much denser than in all other considered instances. Increasing influence probabilities lead to a higher elimination rate but we do not observe values above 16%. For the other real-world and especially the SW instances the influence of $P1$ is negligible.

On the contrary, the conditions for preprocessing $P2$ are easier to satisfy, i.e., if a node is not connected in one particular scenario the corresponding activation variable can be eliminated within this scenario. For $P2$, sparse base graphs and small influence probabilities are beneficial increasing the probability of singleton components up to 80% in our instances. Additionally, the nature of $P2$ makes the elimination rate independent of the number of considered scenarios.

Finally, Fig. 7 compares runtimes for our three solution methods with and without preprocessing, respectively, for both the real-world and the SW instances. The computational advantage of preprocessing in terms of runtime reduction can be clearly observed, most explicitly for the real-world instances and for method BEN.

5. Conclusions

In this work we solve the influence maximization problem based on the independent cascade model by transforming it to a maximal covering location problem with uncertain facility-customer relations. A Benders reformulation of a compact covering formulation leads to a branch-and-cut approach which outperforms the state-of-the-art algorithm (exploiting submodularity) by orders of magnitude in terms of solution runtime and allows to solve much more difficult instances. In particular, two preprocessing steps together with an efficient separation of fractional Benders cuts significantly decrease the solution runtime. To deal with time and memory issues related to large graphs and a high number of scenarios we apply several algorithmic techniques based on incremental calculation and partial storage of reachability sets. Interesting topics for future work could be the development of further techniques to speed-up separation, additional preprocessing procedures and the derivation of stronger cuts (which should still be efficient to separate for large-scale instances). Moreover, it can also be interesting to investigate whether similar branch-and-cut approaches can also be used for other variants of influence maximization problems. To further understand the relation between graph structure and algorithmic performance, one could analyze in more detail the live-arc graphs' properties such as node degree distribution and the number and sizes of (strongly) connected components.

Appendix A. Detailed experimental results

Table A1
Comparison of solution times in seconds and numbers of added Benders and submodular cuts, respectively, of our approaches and state-of-the-art algorithm WK with a stopping criterion of 1% optimality gap. TL denotes “time limit reached”.

name	K	Ω	BEN		BENF		BENFR		WK	
			t[s]	#C	t[s]	#C	t[s]	#C	t[s]	#C
MSG	2	100	2	199	2	199	2	199	8	200
		200	3	400	3	400	3	400	21	400
		300	4	600	4	600	4	600	21	600
		500	6	1000	6	1000	6	1000	37	1000
	3	100	2	201	2	201	2	201	15	208
		200	3	400	3	400	3	400	20	400
		300	4	600	4	600	4	600	22	600
		500	6	1000	8	1000	6	1000	39	1000
	4	100	2	202	2	202	2	202	11	200
		200	3	400	3	400	3	400	14	400
		300	4	608	4	608	4	608	22	600
		500	7	1022	7	1022	7	1022	36	1000
	5	100	2	236	2	236	2	236	7	200
		200	3	400	3	400	3	400	14	400
		300	5	1548	5	620	5	620	22	600
		500	6	1039	6	1039	6	1039	36	1000
ENRON	2	100	72	200	74	200	73	200	1387	200
		200	130	400	131	400	132	400	2757	400
		300	196	600	196	600	195	600	TL	600
		500	314	1000	314	1000	314	1000	TL	1000
	3	100	70	200	72	200	71	200	1374	200
		200	132	400	131	400	131	400	2833	400
		300	188	600	188	600	189	600	TL	600
		500	319	1000	316	1000	311	1000	TL	1000
	4	100	76	200	77	200	75	200	1412	200
		200	139	400	142	400	139	400	2812	400
		300	213	600	212	600	211	600	TL	600
		500	365	1000	362	1000	355	1000	TL	1000
	5	100	77	226	76	226	78	226	1422	200
		200	144	411	142	411	145	411	2870	400
		300	219	600	217	600	220	600	TL	600
		500	362	1000	362	1000	362	1000	TL	1000
GNU	2	100	8	200	8	200	8	200	77	200
		200	15	400	15	400	15	400	149	400
		300	24	600	24	600	24	600	223	600
		500	39	1000	39	1000	40	1000	375	1000
	3	100	8	200	8	200	8	200	73	200
		200	15	400	16	400	16	400	148	400
		300	23	600	23	600	23	600	225	600
		500	41	1000	40	1000	41	1000	380	1000
	4	100	8	200	8	200	8	200	74	200
		200	15	400	16	400	15	400	149	400
		300	23	600	23	600	23	600	223	600
		500	38	1000	38	1000	39	1000	384	1000
	5	100	9	206	9	206	9	206	84	200
		200	20	401	19	415	19	415	161	400
		300	31	600	31	636	30	636	247	600
		500	52	1000	54	1052	54	1052	416	1000
HEP	2	100	7	200	8	200	8	200	112	200
		200	14	400	17	400	15	400	218	400
		300	23	600	23	600	24	600	326	600
		500	39	1000	38	1000	39	1000	553	1000
	3	100	9	279	10	282	10	282	179	279
		200	16	595	16	595	16	595	379	595
		300	23	893	23	893	23	893	580	893
		500	43	1491	43	1491	42	1491	985	1491
	4	100	9	279	8	279	8	279	181	279
		200	20	595	20	595	20	595	414	595
		300	31	893	33	893	33	893	600	893
		500	44	1491	44	1491	44	1491	1008	1491
	5	100	13	320	19	570	20	570	241	322
		200	26	803	40	1135	42	1135	676	793
		300	40	1198	64	1711	62	1711	1067	1187
		500	65	1972	98	2835	101	2835	1753	1996

Table A2

Comparison of solution times in seconds and numbers of added Benders and submodular cuts, respectively, of our approaches and state-of-the-art algorithm WK with a stopping criterion of 0% optimality gap. TL denotes “time limit reached”.

name	K	Ω	BEN		BENF		BENFR		WK	
			t[s]	#C	t[s]	#C	t[s]	#C	t[s]	#C
MSG	2	100	2	211	2	213	2	213	8	200
		200	4	882	4	755	4	755	14	400
		300	5	1398	5	960	5	960	22	600
		500	9	1823	8	1334	8	1334	36	1000
	3	100	2	482	2	391	2	391	11	208
		200	6	2111	3	878	4	878	334	3044
		300	8	3053	5	1034	5	1034	28	632
		500	12	3602	8	1705	8	1705	2006	7975
	4	100	2	352	2	283	2	283	TL	6830
		200	8	3425	3	957	3	957	TL	6249
		300	4	1333	4	889	4	889	TL	9745
		500	19	6741	9	2523	9	2523	TL	9969
	5	100	2	666	2	398	2	398	TL	7709
		200	10	4340	4	1192	4	1192	TL	5855
		300	18	6087	6	663	5	663	TL	9852
		500	16	5705	8	2454	8	2454	TL	9940
ENRON	2	100	80	207	172	208	174	208	1459	207
		200	146	407	375	407	366	407	2826	400
		300	221	605	TL	605	743	605	TL	600
		500	340	1004	1820	1004	1204	1004	TL	1000
	3	100	77	223	216	214	212	214	1691	223
		200	143	416	1633	426	544	421	3102	416
		300	207	617	TL	683	709	614	TL	600
		500	360	1011	2645	1026	1215	1011	TL	1000
	4	100	84	227	177	225	139	223	1837	233
		200	146	416	379	449	401	449	3056	408
		300	267	700	TL	732	749	678	TL	600
		500	462	1171	TL	1114	1520	1102	TL	1000
	5	100	82	239	91	252	91	252	2177	271
		200	150	411	144	411	143	411	2958	416
		300	234	695	1231	738	693	694	TL	600
		500	380	1166	1711	1170	1281	1170	TL	1000
GNU	2	100	29	203	318	204	25	204	113	203
		200	94	405	1310	403	95	403	505	404
		300	250	616	2841	614	228	614	2784	614
		500	2963	1013	TL	1023	1394	1023	TL	1000
	3	100	24	223	91	258	16	217	111	216
		200	338	431	TL	974	193	417	855	423
		300	3092	641	TL	885	2329	620	TL	600
		500	TL	1079	TL	1442	TL	1034	TL	1000
	4	100	12	216	16	228	14	226	100	209
		200	37	448	664	803	50	442	220	426
		300	444	710	TL	1636	175	659	1193	663
		500	TL	1175	TL	3326	TL	1107	TL	1000
	5	100	11	227	14	228	13	228	211	267
		200	33	517	49	480	31	440	367	546
		300	142	782	423	1211	59	686	484	706
		500	1753	1362	TL	2927	698	1175	TL	1260
HEP	2	100	8	200	8	200	8	200	116	200
		200	16	400	14	400	15	400	217	400
		300	23	600	23	600	23	600	334	600
		500	40	1000	39	1000	40	1000	561	1000
	3	100	12	279	10	282	10	282	184	279
		200	18	595	16	595	16	595	383	595
		300	23	893	26	893	23	893	587	893
		500	44	1491	44	1491	41	1491	1005	1491
	4	100	16	375	15	495	15	495	337	371
		200	29	713	30	919	29	919	887	711
		300	47	1048	49	1395	49	1395	845	1033
		500	90	1746	82	2287	79	2287	1567	1736
	5	100	21	347	21	570	19	570	324	347
		200	32	806	41	1135	41	1135	1139	814
		300	55	1220	65	1711	64	1711	1472	1215
		500	77	1992	101	2835	103	2835	2132	1992

Table A3
Detailed results for instance set MSG listing solution times in seconds ($t[s]$), remaining optimality gaps in % ($g[\%]$), primal gaps to the best known solution ($p[\%]$), numbers of added Benders cuts ($\#C$), and numbers of branch-and-bound nodes ($\#B$). TL denotes “time limit reached”.

K	$ \Omega $	p	BEN					BENF					BENFR					WK			
			$t[s]$	$g[\%]$	$p[\%]$	$\#C$	$\#B$	$t[s]$	$g[\%]$	$p[\%]$	$\#C$	$\#B$	$t[s]$	$g[\%]$	$p[\%]$	$\#C$	$\#B$	$t[s]$	$g[\%]$	$p[\%]$	$\#C$
2	250	0.01	2	0.00	0.00	577	7	2	0.00	0.00	569	0	2	0.00	0.00	569	0	17	0.00	0.00	577
		0.05	3	0.00	0.00	923	68	3	0.00	0.00	883	0	3	0.00	0.00	883	0	57	0.00	0.00	947
		0.10	4	0.00	0.00	1271	44	4	0.00	0.00	938	0	4	0.00	0.00	938	0	18	0.00	0.00	500
	500	0.01	3	0.00	0.00	1237	6	3	0.00	0.00	1145	0	3	0.00	0.00	1145	0	40	0.00	0.00	1198
		0.05	7	0.00	0.00	1787	88	6	0.00	0.00	1648	0	6	0.00	0.00	1648	0	94	0.00	0.00	1572
		0.10	9	0.00	0.00	1823	13	8	0.00	0.00	1334	0	8	0.00	0.00	1334	0	36	0.00	0.00	1000
	750	0.01	7	0.00	0.00	1831	12	5	0.00	0.00	1897	3	6	0.00	0.00	1840	6	68	0.00	0.00	1761
		0.05	12	0.00	0.00	2495	19	9	0.00	0.00	2579	0	9	0.00	0.00	2579	0	169	0.00	0.00	2484
		0.10	13	0.00	0.00	1892	20	12	0.00	0.00	1977	0	12	0.00	0.00	1977	0	63	0.00	0.00	1501
	5	0.01	2	0.00	0.00	733	11	2	0.00	0.00	832	3	2	0.00	0.00	832	3	49	0.00	0.00	953
5	250	0.05	7	0.00	0.00	2042	305	3	0.00	0.00	974	0	3	0.00	0.00	974	0	TL	0.08	0.00	7845
		0.10	8	0.00	0.00	2711	266	5	0.00	0.00	537	3	4	0.00	0.00	537	3	45	0.00	0.00	765
		0.01	3	0.00	0.00	1532	43	3	0.00	0.00	1582	3	3	0.00	0.00	1552	7	85	0.00	0.00	1566
	500	0.05	15	0.00	0.00	4285	224	7	0.00	0.00	2269	0	7	0.00	0.00	2269	0	TL	0.22	0.01	2454
		0.10	16	0.00	0.00	5705	344	8	0.00	0.00	2454	0	8	0.00	0.00	2454	0	TL	0.03	0.00	9940
		0.01	5	0.00	0.00	2376	23	5	0.00	0.00	2283	3	5	0.00	0.00	2283	3	153	0.00	0.00	2321
	750	0.05	42	0.00	0.00	8265	439	10	0.00	0.00	3321	0	10	0.00	0.00	3321	0	TL	0.04	0.00	15364
		0.10	53	0.00	0.00	12,684	533	12	0.00	0.00	3785	0	12	0.00	0.00	3785	0	TL	0.32	0.04	2987
	10	0.01	4	0.00	0.00	1471	238	2	0.00	0.00	1194	3	2	0.00	0.00	1194	3	TL	0.25	0.09	5116
		0.05	10	0.00	0.00	2196	321	4	0.00	0.00	1279	0	4	0.00	0.00	1279	0	TL	0.54	0.02	3750
10	250	0.10	2352	0.00	0.00	12,917	39112	4	0.00	0.00	1476	0	4	0.00	0.00	1476	0	TL	0.34	0.00	5746
		0.01	6	0.00	0.00	2633	157	5	0.00	0.00	2633	3	5	0.00	0.00	2633	3	TL	0.33	0.07	8667
		0.05	21	0.00	0.00	5000	322	7	0.00	0.00	2912	0	9	0.00	0.00	2912	0	TL	0.49	0.02	6500
	500	0.10	1080	0.00	0.00	23,515	7861	8	0.00	0.00	3107	0	8	0.00	0.00	3107	0	TL	0.36	0.03	7987
		0.01	9	0.00	0.00	4164	132	6	0.00	0.00	3345	3	6	0.00	0.00	3345	3	TL	0.37	0.04	10824
		0.05	44	0.00	0.00	8747	365	10	0.00	0.00	3447	0	10	0.00	0.00	3447	0	TL	0.27	0.00	15553
	750	0.10	TL	0.01	0.00	37495	12665	14	0.00	0.00	5856	0	14	0.00	0.00	5856	0	TL	0.63	0.11	3745
		0.01	23	0.00	0.00	2682	962	3	0.00	0.00	1806	3	3	0.00	0.00	1804	14	TL	1.10	0.19	3000
		0.05	26	0.00	0.00	4036	631	4	0.00	0.00	2165	3	4	0.00	0.00	2165	3	TL	0.89	0.02	5500
	5	0.10	3266	0.00	0.00	12,510	50517	5	0.00	0.00	1712	0	4	0.00	0.00	1712	0	TL	0.66	0.05	3000
15	250	0.01	52	0.00	0.00	6542	700	6	0.00	0.00	3691	3	7	0.00	0.00	3744	17	TL	1.16	0.18	4497
		0.05	29	0.00	0.00	4761	623	6	0.00	0.00	2571	0	6	0.00	0.00	2571	0	TL	0.92	0.03	7000
		0.10	TL	0.02	0.00	31,250	14825	10	0.00	0.00	4390	0	10	0.00	0.00	4390	0	TL	0.70	0.05	3000
	500	0.01	67	0.00	0.00	8778	657	9	0.00	0.00	4890	3	8	0.00	0.00	4890	3	TL	1.23	0.18	6000
		0.05	85	0.00	0.00	10051	769	11	0.00	0.00	4273	0	10	0.00	0.00	4273	0	TL	0.52	0.00	10500
		0.10	TL	0.02	0.00	40805	9806	15	0.00	0.00	6842	0	15	0.00	0.00	6842	0	TL	0.94	0.11	3749
	750	0.01	TL	0.20	0.02	37334	12038	5	0.00	0.00	3038	5	10	0.00	0.00	2965	254	TL	2.86	0.24	2000
		0.05	45	0.00	0.00	5143	1239	4	0.00	0.00	1885	3	4	0.00	0.00	1885	3	TL	1.78	0.06	3500
		0.10	2284	0.00	0.00	9972	56266	5	0.00	0.00	2920	3	8	0.00	0.00	2926	119	TL	1.29	0.09	3250
	500	0.01	TL	0.27	0.03	49217	4147	10	0.00	0.00	5946	3	143	0.00	0.00	7221	3515	TL	2.62	0.26	4000
		0.05	123	0.00	0.00	7061	2289	8	0.00	0.00	3171	3	8	0.00	0.00	3172	9	TL	1.80	0.08	6000
25	250	0.10	TL	0.01	0.00	22425	20626	9	0.00	0.00	5009	0	9	0.00	0.00	5009	0	TL	1.24	0.12	4500
		0.01	TL	0.26	0.03	73208	4253	14	0.00	0.00	8775	3	192	0.00	0.00	10001	2179	TL	2.61	0.17	5250
		0.05	525	0.00	0.00	23548	2429	13	0.00	0.00	6568	3	24	0.00	0.00	6111	140	TL	1.45	0.04	7500
	500	0.10	TL	0.02	0.00	34562	11786	14	0.00	0.00	7163	0	14	0.00	0.00	7163	0	TL	1.55	0.19	4500

Table A4

Detailed results for instance set HEP listing solution times in seconds ($t[s]$), remaining optimality gaps in % ($g[\%]$), primal gaps to the best known solution ($p[\%]$), numbers of added Benders cuts ($\#C$), and numbers of branch-and-bound nodes ($\#B$). TL denotes “time limit reached”.

K	$ \Omega $	p	BEN					BENF					BENFR					WK			
			$t[s]$	$g[\%]$	$p[\%]$	$\#C$	$\#B$	$t[s]$	$g[\%]$	$p[\%]$	$\#C$	$\#B$	$t[s]$	$g[\%]$	$p[\%]$	$\#C$	$\#B$	$t[s]$	$g[\%]$	$p[\%]$	$\#C$
2	250	0.01	8	0.00	0.00	501	8	8	0.00	0.00	671	0	8	0.00	0.00	671	0	111	0.00	0.00	656
		0.05	10	0.00	0.00	500	0	10	0.00	0.00	500	0	10	0.00	0.00	500	0	127	0.00	0.00	500
		0.10	19	0.00	0.00	500	0	20	0.00	0.00	500	0	19	0.00	0.00	500	0	278	0.00	0.00	500
	500	0.01	15	0.00	0.00	1004	0	15	0.00	0.00	1323	0	15	0.00	0.00	1323	0	153	0.00	0.00	1004
		0.05	22	0.00	0.00	1000	0	20	0.00	0.00	1000	0	20	0.00	0.00	1000	0	242	0.00	0.00	1000
		0.10	41	0.00	0.00	1000	0	40	0.00	0.00	1000	0	41	0.00	0.00	1000	0	563	0.00	0.00	1000
	750	0.01	25	0.00	0.00	1973	0	22	0.00	0.00	1973	0	22	0.00	0.00	1973	0	496	0.00	0.00	1973
		0.05	32	0.00	0.00	1500	0	30	0.00	0.00	1500	0	29	0.00	0.00	1500	0	393	0.00	0.00	1500
		0.10	56	0.00	0.00	1500	0	57	0.00	0.00	1500	0	56	0.00	0.00	1500	0	862	0.00	0.00	1500
		0.01	16	0.00	0.00	1117	171	14	0.00	0.00	1180	0	14	0.00	0.00	1180	0	562	0.00	0.00	1237
5	250	0.05	78	0.00	0.00	999	1301	170	0.00	0.00	1667	261	50	0.00	0.00	1316	384	460	0.00	0.00	976
		0.10	41	0.00	0.00	1014	169	46	0.00	0.00	1430	0	43	0.00	0.00	1430	0	1088	0.00	0.00	1009
		0.01	34	0.00	0.00	2085	344	28	0.00	0.00	2392	3	27	0.00	0.00	2392	3	1304	0.00	0.00	2306
	500	0.05	154	0.00	0.00	1956	745	220	0.00	0.00	3016	115	94	0.00	0.00	2523	145	1377	0.00	0.00	1956
		0.10	78	0.00	0.00	1992	53	103	0.00	0.00	2835	0	100	0.00	0.00	2835	0	2238	0.00	0.00	1992
		0.01	58	0.00	0.00	3975	388	43	0.00	0.00	3871	3	42	0.00	0.00	3746	5	2350	0.00	0.00	3764
	750	0.05	183	0.00	0.00	2934	366	853	0.00	0.00	6169	323	235	0.00	0.00	3911	175	2082	0.00	0.00	2934
		0.10	338	0.00	0.00	3566	93	168	0.00	0.00	4760	6	162	0.00	0.00	4753	8	TL	0.00	0.00	2931
		0.01	TL	5.00	1.15	14,657	6600	27	0.00	0.00	2415	3	27	0.00	0.00	2415	3	TL	6.15	1.79	1913
		0.05	882	0.00	0.00	1951	17,213	1551	0.00	0.00	4753	2313	266	0.00	0.00	2447	2356	TL	0.01	0.00	1288
10	250	0.10	397	0.00	0.00	1862	2234	77	0.00	0.00	1997	0	76	0.00	0.00	1997	0	TL	575.01	8.14	500
		0.01	TL	5.72	1.10	24,149	3583	62	0.00	0.00	4857	3	66	0.00	0.00	4682	7	TL	6.58	1.44	2848
		0.05	TL	0.22	0.04	3765	23,104	TL	0.29	0.08	9819	1102	2182	0.00	0.00	5065	10,790	TL	1.81	1.54	2499
	500	0.10	TL	0.02	0.01	4600	12,792	203	0.00	0.00	3546	23	159	0.00	0.00	3541	24	TL	1.90	1.58	3000
		0.01	TL	7.32	1.98	28,185	2053	115	0.00	0.00	7734	7	113	0.00	0.00	7179	21	TL	8.05	2.27	3622
		0.05	TL	0.20	0.02	4727	11,584	TL	0.26	0.06	11821	1306	1949	0.00	0.00	7532	7366	TL	1.63	1.35	3747
	750	0.10	TL	0.03	0.00	7027	2400	296	0.00	0.00	5959	21	245	0.00	0.00	5959	21	TL	3.31	2.24	3750
		0.01	TL	11.68	2.01	16,664	2800	290	0.00	0.00	10626	170	919	0.00	0.00	4976	11,782	TL	18.60	6.81	1250
		0.05	TL	0.19	0.01	3233	15,700	384	0.00	0.00	4434	415	201	0.00	0.00	3276	1012	TL	1.35	1.12	1500
		0.10	TL	0.18	0.04	3558	9472	158	0.00	0.00	3760	2	154	0.00	0.00	3760	2	TL	573.06	9.58	500
15	250	0.01	TL	13.03	2.77	20,195	2300	621	0.00	0.00	17789	178	2808	0.00	0.00	8860	14,506	TL	28.53	10.52	2000
		0.05	TL	1.36	0.11	8798	3602	TL	0.09	0.00	12014	1917	1119	0.00	0.00	6199	4536	TL	7.27	5.63	2000
		0.10	TL	3.68	0.14	7204	1635	1341	0.00	0.00	6400	146	1123	0.00	0.00	6041	345	TL	572.74	9.47	1000
	500	0.01	TL	13.06	2.25	27208	1493	1582	0.00	0.00	31945	209	TL	0.63	0.10	22459	8240	TL	26.20	8.88	3000
		0.05	TL	1.36	0.00	9633	2500	TL	0.20	0.04	14743	551	TL	0.05	0.00	9354	3571	TL	7.34	5.61	3000
		0.10	TL	1.70	0.08	7393	1242	2901	0.00	0.00	9210	334	1512	0.00	0.00	8964	377	TL	574.31	9.61	1500
	750	0.01	TL	17.59	3.98	14458	2565	2288	0.00	0.00	26086	380	TL	0.32	0.04	7271	19741	TL	34.10	11.54	1000
		0.05	TL	2.41	0.40	13726	1700	1024	0.00	0.00	7334	796	1837	0.00	0.00	5173	10245	TL	1207.16	10.92	500
		0.10	TL	5.73	0.56	9910	1713	411	0.00	0.00	4769	51	401	0.00	0.00	4607	261	TL	568.51	11.46	500
		0.01	TL	18.02	4.03	24625	1325	TL	0.43	0.00	41751	199	TL	0.88	0.25	17098	5577	TL	31.07	9.46	2000
25	250	0.05	TL	25.12	0.53	16051	1060	TL	0.23	0.00	18464	344	TL	0.22	0.00	10676	2666	TL	1234.85	12.47	1000
		0.10	TL	8.83	0.26	9813	815	2919	0.00	0.00	9696	452	TL	0.02	0.01	8867	1800	TL	570.17	11.60	1000
		0.01	TL	17.65	3.31	26941	1115	TL	0.20	0.00	43852	188	TL	0.89	0.26	17486	1990	TL	34.15	11.41	3000
	500	0.05	TL	25.04	0.72	17386	766	TL	0.24	0.00	25379	107	TL	0.22	0.00	13336	737	TL	1231.34	12.09	1500
		0.10	TL	9.33	0.83	12972	704	TL	8.44	0.01	17847	110	TL	0.02	0.00	15583	558	TL	563.15	10.59	1500

Table A5
Detailed results for instance set GNU listing solution times in seconds (t[s]), remaining optimality gaps in % (g[%]), primal gaps to the best known solution (p[%]), numbers of added Benders cuts (#C), and numbers of branch-and-bound nodes (#B). TL denotes “time limit reached”.

K	Ω	p	BEN					BENF					BENFR					WK			
			t[s]	g[%]	p[%]	#C	#B	t[s]	g[%]	p[%]	#C	#B	t[s]	g[%]	p[%]	#C	#B	t[s]	g[%]	p[%]	#C
2	250	0.01	4	0.00	0.00	252	0	4	0.00	0.00	252	0	4	0.00	0.00	252	0	10	0.00	0.00	252
		0.05	7	0.00	0.00	416	0	7	0.00	0.00	451	0	7	0.00	0.00	451	0	38	0.00	0.00	416
		0.10	114	0.00	0.00	507	1089	1775	0.00	0.00	505	1061	118	0.00	0.00	505	1061	444	0.00	0.00	505
	500	0.01	8	0.00	0.00	502	0	8	0.00	0.00	502	0	8	0.00	0.00	502	0	24	0.00	0.00	502
		0.05	19	0.00	0.00	950	0	17	0.00	0.00	965	0	17	0.00	0.00	965	0	139	0.00	0.00	950
		0.10	2538	0.00	0.00	1013	4422	TL	0.11	0.04	1023	553	1705	0.00	0.00	1023	3101	TL	98.22	0.14	1000
	750	0.01	14	0.00	0.00	753	0	12	0.00	0.00	753	0	12	0.00	0.00	753	0	43	0.00	0.00	753
		0.05	31	0.00	0.00	1357	6	32	0.00	0.00	1605	2	30	0.00	0.00	1605	2	257	0.00	0.00	1357
		0.10	TL	0.07	0.00	1507	3831	TL	0.13	0.04	1521	246	TL	0.04	0.00	1521	2222	TL	98.19	0.12	1500
	5	0.01	5	0.00	0.00	294	0	5	0.00	0.00	305	0	5	0.00	0.00	305	0	21	0.00	0.00	294
5	250	0.05	118	0.00	0.00	4660	1181	32	0.00	0.00	2325	19	28	0.00	0.00	1955	92	TL	0.07	0.00	3658
		0.10	135	0.00	0.00	647	1963	589	0.00	0.00	1066	835	49	0.00	0.00	596	144	511	0.00	0.00	652
		0.01	11	0.00	0.00	650	0	11	0.00	0.00	671	0	10	0.00	0.00	671	0	147	0.00	0.00	662
	500	0.05	292	0.00	0.00	8468	1309	85	0.00	0.00	5624	26	60	0.00	0.00	2922	148	TL	1.34	0.29	5543
		0.10	1722	0.00	0.00	1362	17,240	TL	0.06	0.04	2921	2747	683	0.00	0.00	1175	13,648	TL	0.02	0.00	1170
		0.01	16	0.00	0.00	948	0	16	0.00	0.00	971	0	16	0.00	0.00	971	0	296	0.00	0.00	1061
	750	0.05	632	0.00	0.00	15,624	1159	101	0.00	0.00	6581	19	84	0.00	0.00	4169	135	TL	2.96	0.96	6523
		0.10	TL	0.14	0.03	1841	7248	TL	0.09	0.03	3135	832	TL	0.05	0.00	1761	19,343	TL	321.24	0.56	1500
	10	0.01	45	0.00	0.00	4249	1617	9	0.00	0.00	1024	0	9	0.00	0.00	1024	0	TL	0.45	0.00	3015
		0.05	TL	13.95	2.34	14,832	2130	719	0.00	0.00	14,980	227	TL	0.47	0.00	6351	28,910	TL	16.18	3.04	1243
10	250	0.10	108	0.00	0.00	1451	2886	46	0.00	0.00	909	4	50	0.00	0.00	928	24	TL	0.22	0.03	2683
		0.01	214	0.00	0.00	12793	3383	18	0.00	0.00	2087	0	18	0.00	0.00	2087	0	TL	0.80	0.11	3667
		0.05	TL	17.29	3.45	24,249	1228	TL	0.26	0.00	31,892	326	TL	2.39	0.54	17196	2785	TL	19.34	4.38	1991
	500	0.10	324	0.00	0.00	1775	2079	74	0.00	0.00	1479	4	90	0.00	0.00	1513	219	TL	0.14	0.01	3959
		0.01	82	0.00	0.00	9096	962	23	0.00	0.00	2173	0	23	0.00	0.00	2173	0	TL	0.60	0.00	4445
		0.05	TL	15.82	2.61	26,384	1056	1249	0.00	0.00	30,295	161	TL	1.01	0.00	13,916	9551	TL	18.31	3.07	2975
	750	0.10	TL	0.01	0.00	8228	6000	123	0.00	0.00	2710	4	159	0.00	0.00	2654	100	TL	0.06	0.00	3971
		0.01	TL	0.99	0.08	61,044	10,000	14	0.00	0.00	2232	0	13	0.00	0.00	2232	0	TL	1.41	0.08	3637
		0.05	TL	20.12	3.74	12,605	2033	TL	1.03	0.00	24,089	249	TL	1.70	0.18	7077	7150	TL	25.08	5.64	1000
	5	0.10	TL	0.04	0.00	8616	7987	40	0.00	0.00	1335	7	37	0.00	0.00	1288	14	TL	0.15	0.01	2239
15	250	0.01	TL	1.34	0.06	88,278	5746	19	0.00	0.00	2682	0	20	0.00	0.00	2682	0	TL	1.69	0.19	4914
		0.05	TL	18.92	3.52	23,165	1300	TL	0.44	0.00	33,299	210	TL	1.37	0.00	8873	6370	TL	23.14	4.52	1999
		0.10	TL	0.02	0.00	7255	6500	91	0.00	0.00	2344	7	1355	0.00	0.00	3089	4319	TL	0.09	0.03	2401
	500	0.01	TL	0.89	0.00	104,068	6695	29	0.00	0.00	3944	0	27	0.00	0.00	3944	0	TL	1.47	0.14	5721
		0.05	TL	22.40	4.28	22060	947	TL	1.61	0.05	36,161	98	TL	2.02	0.00	18632	2240	TL	30.66	4.28	2248
		0.10	TL	0.59	0.03	26796	1252	188	0.00	0.00	3316	1	180	0.00	0.00	3316	1	TL	0.07	0.01	3115
	750	0.01	TL	2.99	0.18	27703	1950	21	0.00	0.00	4063	0	20	0.00	0.00	4063	0	TL	3.15	0.24	3640
		0.05	TL	23.13	2.85	13295	1700	TL	2.57	0.00	25,571	120	TL	2.82	0.03	11162	2527	TL	36.78	4.78	750
		0.10	TL	0.15	0.03	11431	3804	53	0.00	0.00	2069	10	75	0.00	0.00	2112	379	TL	1.62	0.27	1500
	5	0.01	TL	2.68	0.08	42336	1368	39	0.00	0.00	7130	0	36	0.00	0.00	7130	0	TL	3.10	0.36	4985
		0.05	TL	25.03	3.82	20748	1000	TL	4.03	1.29	27,665	66	TL	2.72	0.00	15870	972	TL	36.46	4.88	1500
25	250	0.10	TL	1.32	0.13	21083	1465	293	0.00	0.00	6050	39	1634	0.00	0.00	5981	4818	TL	0.21	0.03	2460
		0.01	TL	2.89	0.34	49376	980	46	0.00	0.00	9722	0	43	0.00	0.00	9722	0	TL	2.99	0.34	5801
		0.05	TL	29.39	5.00	23495	710	TL	2.29	0.00	33577	47	TL	2.46	0.00	15587	430	TL	36.59	5.55	2250
	500	0.10	TL	0.11	0.01	22593	1700	1167	0.00	0.00	7449	158	TL	0.03	0.01	16409	4734	TL	0.17	0.03	3154

Table A6

Detailed results for instance set ENRON listing solution times in seconds ($t[s]$), remaining optimality gaps in % ($g[\%]$), primal gaps to the best known solution ($p[\%]$), numbers of added Benders cuts ($\#C$), and numbers of branch-and-bound nodes ($\#B$). TL denotes “time limit reached”. Entries “-” for WK indicate that the initial formulation could not be built within the time limit.

K	Ω	p	BEN					BENF					BENFR					WK			
			t[s]	g[%]	p[%]	#C	#B	t[s]	g[%]	p[%]	#C	#B	t[s]	g[%]	p[%]	#C	#B	t[s]	g[%]	p[%]	#C
2	250	0.01	35	0.00	0.00	500	0	32	0.00	0.00	500	0	32	0.00	0.00	500	0	407	0.00	0.00	500
		0.05	96	0.00	0.00	504	0	90	0.00	0.00	504	0	90	0.00	0.00	504	0	2036	0.00	0.00	502
		0.10	174	0.00	0.00	504	31	599	0.00	0.00	504	4	606	0.00	0.00	504	4	TL	100.00	0.10	500
	500	0.01	66	0.00	0.00	1001	0	61	0.00	0.00	1001	0	61	0.00	0.00	1001	0	985	0.00	0.00	1001
		0.05	180	0.00	0.00	1003	0	196	0.00	0.00	1003	0	196	0.00	0.00	1003	0	TL	100.00	0.36	1000
		0.10	341	0.00	0.00	1004	18	1703	0.00	0.00	1004	9	1193	0.00	0.00	1004	9	TL	100.00	0.10	1000
	750	0.01	102	0.00	0.00	1501	0	96	0.00	0.00	1501	0	95	0.00	0.00	1501	0	1551	0.00	0.00	1501
		0.05	255	0.00	0.00	1508	0	254	0.00	0.00	1508	0	261	0.00	0.00	1508	0	TL	100.00	0.37	1500
		0.10	689	0.00	0.00	1503	59	1416	0.00	0.00	1503	0	1441	0.00	0.00	1503	0	TL	-	-	-
	5	250	0.01	45	0.00	0.00	524	244	201	0.00	0.00	668	214	54	0.00	0.00	550	186	718	0.00	0.00
0.05			105	0.00	0.00	648	27	252	0.00	0.00	662	16	134	0.00	0.00	653	16	3016	0.00	0.00	648
0.10			199	0.00	0.00	606	82	2557	0.00	0.00	649	38	909	0.00	0.00	591	36	TL	256.69	0.37	500
500		0.01	78	0.00	0.00	1030	172	270	0.00	0.00	1196	114	99	0.00	0.00	1128	96	1182	0.00	0.00	1015
		0.05	188	0.00	0.00	1286	4	285	0.00	0.00	1454	3	274	0.00	0.00	1454	3	TL	399.99	0.80	1000
		0.10	383	0.00	0.00	1166	12	1749	0.00	0.00	1170	10	1276	0.00	0.00	1170	10	TL	256.88	0.39	1000
750		0.01	129	0.00	0.00	1529	198	1437	0.00	0.00	2263	524	282	0.00	0.00	1673	192	1892	0.00	0.00	1523
		0.05	283	0.00	0.00	1925	4	374	0.00	0.00	1926	3	372	0.00	0.00	1926	3	TL	399.99	0.80	1500
		0.10	654	0.00	0.00	1674	47	TL	0.01	0.00	2105	5	2492	0.00	0.00	2096	32	TL	-	-	-
10		250	0.01	35	0.00	0.00	604	25	36	0.00	0.00	628	0	37	0.00	0.00	628	0	623	0.00	0.00
	0.05		141	0.00	0.00	1295	153	143	0.00	0.00	1252	0	148	0.00	0.00	1252	0	TL	0.06	0.03	885
	0.10		202	0.00	0.00	724	27	245	0.00	0.00	847	0	249	0.00	0.00	847	0	TL	0.41	0.22	750
	500	0.01	95	0.00	0.00	1309	173	190	0.00	0.00	1810	95	96	0.00	0.00	1306	96	2242	0.00	0.00	1246
		0.05	409	0.00	0.00	3446	345	529	0.00	0.00	2561	26	375	0.00	0.00	2436	25	TL	477.12	1.30	1000
		0.10	438	0.00	0.00	1617	90	890	0.00	0.00	2069	4	906	0.00	0.00	2069	4	TL	256.86	0.72	1000
	750	0.01	203	0.00	0.00	2308	394	265	0.00	0.00	2596	87	155	0.00	0.00	2098	231	TL	0.01	0.00	1864
		0.05	972	0.00	0.00	5112	718	1104	0.00	0.00	4184	25	655	0.00	0.00	3655	97	TL	476.69	1.24	1500
		0.10	784	0.00	0.00	2774	124	1854	0.00	0.00	3442	26	1342	0.00	0.00	3321	24	TL	-	-	-
	15	250	0.01	45	0.00	0.00	1325	286	41	0.00	0.00	944	0	39	0.00	0.00	944	0	TL	0.00	0.00
0.05			TL	0.04	0.00	10,453	2135	200	0.00	0.00	2294	10	185	0.00	0.00	2294	10	TL	0.69	0.28	750
0.10			268	0.00	0.00	1674	1007	325	0.00	0.00	1723	4	319	0.00	0.00	1723	4	TL	256.56	0.90	500
500		0.01	82	0.00	0.00	1938	96	80	0.00	0.00	1649	0	73	0.00	0.00	1649	0	TL	0.01	0.01	1728
		0.05	TL	0.06	0.02	16,511	849	388	0.00	0.00	3724	4	378	0.00	0.00	3724	4	TL	476.78	1.62	1000
		0.10	561	0.00	0.00	3064	1302	604	0.00	0.00	3361	4	633	0.00	0.00	3361	4	TL	256.79	0.94	1000
750		0.01	120	0.00	0.00	2462	94	118	0.00	0.00	2652	0	114	0.00	0.00	2652	0	TL	0.01	0.00	2204
		0.05	TL	0.03	0.01	17057	750	561	0.00	0.00	5273	3	582	0.00	0.00	5273	3	TL	476.91	1.68	1500
		0.10	1034	0.00	0.00	4654	1169	1094	0.00	0.00	5218	0	1046	0.00	0.00	5218	0	TL	-	-	-
25		250	0.01	TL	0.02	0.00	12897	7861	53	0.00	0.00	1943	3	52	0.00	0.00	1890	14	TL	0.58	0.08
	0.05		TL	0.16	0.04	15228	1112	259	0.00	0.00	3906	3	263	0.00	0.00	3902	5	TL	1.44	0.60	750
	0.10		TL	0.05	0.02	9250	1538	389	0.00	0.00	3343	6	364	0.00	0.00	3255	13	TL	256.42	1.28	500
	500	0.01	TL	0.04	0.01	26741	2571	102	0.00	0.00	3911	3	128	0.00	0.00	4297	167	TL	0.30	0.01	2476
		0.05	TL	0.15	0.03	17602	690	581	0.00	0.00	8014	5	573	0.00	0.00	8014	5	TL	476.00	2.09	1000
		0.10	TL	0.12	0.03	15781	458	973	0.00	0.00	5689	9	868	0.00	0.00	5464	13	TL	256.83	1.36	1000
	750	0.01	TL	0.06	0.01	25731	1201	149	0.00	0.00	6021	3	226	0.00	0.00	5817	210	TL	0.60	0.01	2983
		0.05	TL	0.12	0.02	18687	386	892	0.00	0.00	11451	3	865	0.00	0.00	11451	3	TL	476.02	2.12	1500
		0.10	TL	0.10	0.01	20942	245	1606	0.00	0.00	10033	3	1507	0.00	0.00	10030	6	TL	-	-	-

Table A7
Detailed results for SW instances with average node degree 8, listing avg. solution times in seconds ($\bar{t}[s]$), avg. remaining optimality gaps in % ($\bar{g}[\%]$), avg. numbers of added Benders cuts ($\#C$), and avg. numbers of branch-and-bound nodes ($\#B$). Each row corresponds to a set of five instances. TL denotes “time limit reached”.

K	Ω	p	BEN				BENF				BENFR				WK		
			$\bar{t}[s]$	$\bar{g}[\%]$	#C	#B	$\bar{t}[s]$	$\bar{g}[\%]$	#C	#B	$\bar{t}[s]$	$\bar{g}[\%]$	#C	#B	$\bar{t}[s]$	$\bar{g}[\%]$	#C
10	250	β	2088	0.75	17,362	12,980	4	0.00	1870	3	5	0.00	1834	5	TL	1.38	4055
		0.01	1	0.00	260	0	1	0.00	260	0	1	0.00	260	0	10	0.00	294
		0.05	2	0.00	514	79	1	0.00	509	0	1	0.00	509	0	897	0.00	1658
		0.10	TL	1.58	22,919	14,910	6	0.00	1953	12	6	0.00	1768	37	TL	1.65	3371
		β	3338	1.22	37,528	11,254	7	0.00	3553	3	8	0.00	3504	15	TL	1.77	5758
	500	0.01	1	0.00	553	0	1	0.00	553	0	1	0.00	553	0	39	0.00	730
		0.05	3	0.00	895	152	3	0.00	956	0	3	0.00	956	0	795	0.00	2506
		0.10	TL	1.35	34,495	11,105	11	0.00	3516	9	13	0.00	3387	83	TL	1.59	4780
		β	3390	1.58	54,354	6628	14	0.00	5160	8	13	0.00	4880	16	TL	1.95	7263
		0.01	2	0.00	802	0	2	0.00	802	0	2	0.00	802	0	93	0.00	1150
	750	0.05	5	0.00	1299	102	4	0.00	1392	0	4	0.00	1392	0	2459	0.01	4929
		0.10	TL	1.73	47,851	7401	25	0.00	5293	31	64	0.00	5150	881	TL	1.78	5870
	15	β	TL	5.48	35,686	6399	19	0.00	3576	19	46	0.00	2879	793	TL	5.60	2146
		0.01	1	0.00	302	4	1	0.00	311	1	1	0.00	311	1	1502	0.02	2262
		0.05	109	0.00	2957	8148	2	0.00	1138	0	2	0.00	1138	0	TL	0.34	3846
		0.10	TL	5.51	30,340	6776	108	0.00	6403	128	692	0.00	4020	8926	TL	5.60	1970
	500	β	TL	5.20	48,603	3657	35	0.00	6386	21	52	0.00	4861	238	TL	5.75	3279
		0.01	2	0.00	596	1	2	0.00	632	0	2	0.00	632	0	2958	0.14	5096
		0.05	1134	0.02	7859	24,863	4	0.00	1920	0	4	0.00	1920	0	TL	0.43	5397
		0.10	TL	5.11	39,002	5118	773	0.06	13,543	251	808	0.20	8723	2426	TL	5.10	2722
	750	β	TL	5.80	59,317	2968	52	0.00	8622	19	70	0.00	7063	134	TL	5.33	4288
		0.01	2	0.00	917	5	2	0.00	950	0	2	0.00	950	0	2510	0.07	4453
		0.05	572	0.00	9332	15,772	6	0.00	2816	0	7	0.00	2816	0	TL	0.46	7005
		0.10	TL	5.08	52,384	3834	393	0.00	15,561	151	2106	0.15	12,343	9168	TL	5.36	3834
	25	β	TL	11.39	34,219	5361	1580	0.22	22,268	211	3275	0.35	6756	20,077	TL	13.83	1143
		0.01	4	0.00	698	875	1	0.00	616	0	1	0.00	616	0	TL	0.40	6114
		0.05	TL	1.60	38,307	6443	5	0.00	2844	3	5	0.00	2831	8	TL	2.36	3414
		0.10	TL	10.86	28,183	6193	TL	1.28	35,681	279	TL	1.58	11,031	11678	TL	14.62	1037
	500	β	TL	11.33	49,545	3480	2702	0.46	36,342	187	TL	0.79	14,666	7572	TL	13.77	1991
		0.01	20	0.00	1323	4288	2	0.00	1108	0	2	0.00	1108	0	TL	0.56	7667
		0.05	TL	1.57	49,813	4110	9	0.00	5257	1	10	0.00	5257	1	TL	2.12	4642
		0.10	TL	10.03	36,043	3930	TL	2.42	38417	167	TL	2.22	19619	5291	TL	12.46	1872
	750	β	TL	10.87	56,315	2829	3089	0.53	42,854	141	TL	0.67	16,720	3708	TL	15.81	2689
		0.01	19	0.00	1962	1922	3	0.00	1549	1	3	0.00	1549	1	TL	0.58	8892
		0.05	TL	1.50	57155	3158	13	0.00	6885	1	13	0.00	6885	1	TL	2.01	5610
		0.10	TL	10.56	45887	2797	TL	2.86	41729	116	TL	2.59	27502	2869	TL	12.72	2660

Table A8

Detailed results for SW instances with average node degree 16, listing avg. solution times in seconds ($\bar{t}$ [s]), avg. remaining optimality gaps in % ($\bar{g}$ [%]), avg. numbers of added Benders cuts (#C), and avg. numbers of branch-and-bound nodes (#B). Each row corresponds to a set of five instances. TL denotes “time limit reached”.

K	Ω	p	BEN				BENF				BENFR				WK			
			$\bar{t}$ [s]	$\bar{g}$ [%]	#C	#B	$\bar{t}$ [s]	$\bar{g}$ [%]	#C	#B	$\bar{t}$ [s]	$\bar{g}$ [%]	#C	#B	$\bar{t}$ [s]	$\bar{g}$ [%]	#C	
10	250	β	926	0.00	6804	11744	5	0.00	2057	6	5	0.00	2058	41	TL	0.16	3134	
		0.01	1	0.00	370	139	1	0.00	357	0	1	0.00	357	0	2119	0.05	2578	
		0.05	TL	6.04	21,073	10,969	1869	0.41	22,290	640	3308	1.23	9330	30,518	TL	6.19	1549	
	500	0.10	1556	0.00	5401	29,261	4	0.00	1729	1	4	0.00	1729	1	TL	0.18	2667	
		β	2068	0.02	12,012	12,878	9	0.00	4190	4	14	0.00	4147	121	TL	0.32	4234	
		0.01	2	0.00	620	22	2	0.00	647	0	2	0.00	647	0	879	0.00	2093	
	750	0.05	TL	6.68	28,225	7110	TL	1.31	37,959	711	TL	2.26	15,841	14,208	TL	7.62	1886	
		0.10	2859	0.03	12,648	18,161	10	0.00	3719	6	10	0.00	3685	16	TL	0.34	3636	
		β	3322	0.05	24,674	9721	16	0.00	6643	5	45	0.00	6980	325	TL	0.39	5481	
	15	250	0.01	3	0.00	935	34	3	0.00	956	0	3	0.00	956	0	1495	0.00	2979
			0.05	TL	7.68	34,777	4488	TL	2.65	44,292	358	TL	3.16	25,780	6934	TL	7.29	2770
			0.10	TL	0.05	19,238	14,090	17	0.00	5029	10	18	0.00	4934	59	TL	0.38	4915
500		β	TL	0.22	17,116	18,145	5	0.00	2162	5	5	0.00	2139	13	TL	0.78	1770	
		0.01	17	0.00	1026	2915	2	0.00	609	0	2	0.00	609	0	3309	0.25	3859	
		0.05	TL	12.72	18,115	7661	TL	4.10	35,572	275	TL	3.94	11,155	11,757	TL	14.83	997	
750		0.10	TL	0.22	14,295	20434	6	0.00	2052	8	6	0.00	2011	15	TL	0.68	1560	
		β	TL	0.33	29,932	8375	9	0.00	3933	4	10	0.00	3912	9	TL	0.88	2956	
		0.01	8	0.00	1364	1249	3	0.00	1083	0	3	0.00	1083	0	TL	0.23	6151	
25		250	0.05	TL	12.39	25,268	4840	TL	6.48	32,090	97	TL	5.23	17,107	4791	TL	16.89	1610
			0.10	TL	0.36	26,515	9682	21	0.00	4291	44	22	0.00	3804	278	TL	0.90	2570
			β	TL	0.44	38,202	5862	17	0.00	5976	13	18	0.00	5772	57	TL	0.96	3855
	500	0.01	20	0.00	2039	3422	4	0.00	1532	0	4	0.00	1532	0	TL	0.20	6734	
		0.05	TL	12.68	32,180	3127	TL	8.06	33,507	71	TL	6.24	21,023	2854	TL	16.61	2152	
		0.10	TL	0.38	32,407	6966	36	0.00	6411	53	41	0.00	5602	325	TL	0.93	3712	
	750	β	TL	0.89	23954	11,487	11	0.00	3095	23	14	0.00	2900	115	TL	1.70	1147	
		0.01	TL	0.50	23,431	42,032	3	0.00	1840	1	3	0.00	1838	1	TL	0.96	4110	
		0.05	TL	19.82	17,887	5471	TL	7.12	26,599	52	TL	6.63	11878	4151	TL	30.25	749	
	25	250	0.10	TL	0.88	17,708	11,992	312	0.00	6383	620	1014	0.00	3188	21827	TL	1.85	1043
			β	TL	1.13	30,181	5443	29	0.00	5647	26	40	0.00	5222	165	TL	2.09	1994
			0.01	TL	0.60	37,009	8901	7	0.00	3210	7	7	0.00	3111	19	TL	1.08	6613
500		0.05	TL	20.52	25,550	3028	TL	15.28	24,881	27	TL	8.73	19,012	1752	TL	30.96	1396	
		0.10	TL	1.02	26,661	4891	835	0.00	12,438	1074	2520	0.01	5355	26,615	TL	2.25	1994	
		β	TL	1.21	37,604	3450	48	0.00	8044	27	57	0.00	7258	125	TL	1.97	2991	
750		0.01	TL	0.48	40,315	8274	9	0.00	4217	4	8	0.00	4140	7	TL	0.98	7567	
		0.05	TL	20.55	31,485	2226	TL	20.24	23,668	16	TL	9.38	19,664	512	TL	33.88	1947	
		0.10	TL	1.13	33,723	2671	1853	0.00	18,360	1132	3097	0.03	8083	16,205	TL	2.22	2993	

References

- Barabási, A.-L., & Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286(5439), 509–512.
- Bharathi, S., Kempe, D., & Salek, M. (2007). Competitive influence maximization in social networks. In *International workshop on web and internet economics* (pp. 306–311). Springer.
- Chen, W., Lakshmanan, L. V., & Castillo, C. (2013). Information and influence propagation in social networks. *Synthesis Lectures on Data Management*, 5(4), 1–177.
- Chen, W., Wang, Y., & Yang, S. (2009). Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 199–208). ACM.
- Church, R., & Velle, C. R. (1974). The maximal covering location problem. *Papers in Regional Science*, 32(1), 101–118.
- Cordeau, J.-F., Furini, F., & Ljubić, I. (2019). Benders decomposition for very large scale partial set covering and maximal covering location problems. *European Journal of Operational Research*, 275(3), 882–896.
- Cornuéjols, G., Nemhauser, G., & Wolsey, L. (1983). The uncapacitated facility location problem. *Technical Report*. Cornell University Operations Research and Industrial Engineering.
- Dreyer, P. A., Jr., & Roberts, F. S. (2009). Irreversible k-threshold processes: Graph-theoretical threshold models of the spread of disease and of opinion. *Discrete Applied Mathematics*, 157(7), 1615–1627.
- Erdős, P., & Rényi, A. (1960). On the evolution of random graphs. In *Publ. math. inst. hung. acad. sci.* 5 (pp. 17–60).
- Filippi, C., Guastaroba, G., & Speranza, M. G. (2019). On single-source capacitated facility location with cost and fairness objectives. *European Journal of Operational Research*.
- Fischetti, M., Kahr, M., Leitner, M., Monaci, M., & Ruthmair, M. (2018). Least cost influence propagation in (social) networks. *Mathematical Programming*, 170(1), 293–325.
- Fischetti, M., Ljubić, I., & Sinnl, M. (2016a). Benders decomposition without separability: A computational study for capacitated facility location problems. *European Journal of Operational Research*, 253(3), 557–569.
- Fischetti, M., Ljubić, I., & Sinnl, M. (2016b). Redesigning Benders decomposition for large-scale facility location. *Management Science*, 63(7), 2146–2162.
- Gibbs, D. L., & Shmulevich, I. (2017). Solving the influence maximization problem reveals regulatory organization of the yeast cell cycle. *PLoS Computational Biology*, 13(6), e1005591.
- Granovetter, M. (1978). Threshold models of collective behavior. *The American Journal of Sociology*, 83(6), 1420–1443.
- Güney, E. (2017). On the optimal solution of budgeted influence maximization problem in social networks. *Operational Research*, 19(3), 817–831.
- Güney, E. (2019). An efficient linear programming based method for the influence maximization problem in social networks. *Information Sciences*, 503, 589–605.
- Jo, K., Jung, I., Moon, J. H., & Kim, S. (2016). Influence maximization in time bounded network identifies transcription factors regulating perturbed pathways. *Bioinformatics*, 32(12), i128–i136.
- Kempe, D., Kleinberg, J., & 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* (pp. 137–146). ACM.
- Kempe, D., Kleinberg, J., & Tardos, É. (2015). Maximizing the spread of influence through a social network. *Theory of Computing*, 11(4), 105–147.
- Lee, J., Hasenbein, J. J., & Morton, D. P. (2015). Optimization of stochastic virus detection in contact networks. *Operations Research Letters*, 43(1), 59–64.
- Leskovec, J., Lang, K. J., Dasgupta, A., & Mahoney, M. W. (2009). Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1), 29–123.
- Li, X., Smith, J. D., Dinh, T. N., & Thai, M. T. (2019). TipTop: (Almost) exact solutions for influence maximization in billion-scale networks. *IEEE/ACM Transactions on Networking*, 27(2), 649–661.
- Li, Y., Fan, J., Wang, Y., & Tan, K.-L. (2018). Influence maximization on social graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 30(10), 1852–1872.
- Mossel, E., & Roch, S. (2010). Submodularity of influence in social networks: From local to global. *SIAM Journal on Computing*, 39(6), 2176–2188.
- Nemhauser, G., & Wolsey, L. (1981). Maximizing submodular set functions: formulations and analysis of algorithms. In *North-Holland mathematics studies: 59* (pp. 279–301). Elsevier.
- Nemhauser, G., Wolsey, L., & Fisher, M. (1978). An analysis of approximations for maximizing submodular set functions. *Mathematical Programming*, 14(1), 265–294.
- Newman, M. E. (2002). Spread of epidemic disease on networks. *Physical Review E*, 66(1), 016128.
- Panzarasa, P., Opsahl, T., & Carley, K. M. (2009). Patterns and dynamics of users' behavior and interaction: Network analysis of an online community. *Journal of the American Society for Information Science and Technology*, 60(5), 911–932.
- Pastor-Satorras, R., & Vespignani, A. (2001). Epidemic spreading in scale-free networks. *Physical Review Letters*, 86(14), 3200.
- Rahmaniani, R., Crainic, T. G., Gendreau, M., & Rei, W. (2017). The Benders decomposition algorithm: A literature review. *European Journal of Operational Research*, 259(3), 801–817.
- Ripeanu, M., & Foster, I. (2002). Mapping the gnutella network: Macroscopic properties of large-scale peer-to-peer systems. In *International workshop on peer-to-peer systems* (pp. 85–93). Springer.
- Vatsa, A. K., & Jayaswal, S. (2016). A new formulation and Benders decomposition for the multi-period maximal covering facility location problem with server uncertainty. *European Journal of Operational Research*, 251(2), 404–418.
- Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of 'small-world'-networks. *Nature*, 393(6684), 440–442.
- Wentges, P. (1996). Accelerating Benders' decomposition for the capacitated facility location problem. *Mathematical Methods of Operations Research*, 44(2), 267–290.
- Wu, H.-H., & Küçükyavuz, S. (2018). A two-stage stochastic programming approach for influence maximization in social networks. *Computational Optimization and Applications*, 69(3), 563–595.