Managing Conflicting Interests of Stakeholders in Influencer Marketing

ABSTRACT

A successful campaign should be able to attract investment from the *brand*, and meanwhile manage the conflicting interests in the campaign cost between the brand and the *influencers*. As such, the *agency* between these two stakeholders plays a vital role. Motivated by the above, we stand in the agency's shoes to formulate an interesting yet practical problem, namely *Profit Divergence Minimization in Investment-Persuasive Influencer Marketing Campaign* (PDMIC). This problem aims to (i) minimize the *divergence* of the actual hiring prices from the asking prices of the influencers and meanwhile (ii) maintain the attractiveness of the pricing scheme for the influencers to the brand. We show that the PDMIC problem is NP-hard. To mitigate the challenge of the extremely large searching space of the hiring prices of the influencers, we solve this problem by firstly considering a *restrictive* searching sub-space and then gradually expanding the searching sub-space to the whole space in the end (specifically, from binary price choices to a set of integer prices and then to any price in the feasible price range). We propose effective yet efficient (approximate) algorithms for solving the problem in each of these settings. Extensive experiments on real-world datasets demonstrate the superiority of our methods.

ACM Reference Format:

. 2018. Managing Conflicting Interests of Stakeholders in Influencer Marketing. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX)*. ACM, New York, NY, USA, 19 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Influencer marketing, which involves collaborations among the brand, the agency and online influencers to market products, has now become a mainstream form of online marketing and is forecast to notch $15 billion by the end of 2022 [12]. Figure 1 depicts a general workflow of initiating a marketing campaign in influencer marketing platforms such as SocialPubli [7] and Fourth Floor Creative [19]. Given a budget offered by the brand (in Step 1 of Figure 1), the agency first finds candidate influencers and collects the asking prices (in Step 2). Then in Step 3, the agency makes appropriate prices for hiring (some of) these influencers, returns the individual hiring price to each influencer, and returns the whole pricing scheme to the brand. Finally, the two stakeholders respond to the plan (in Step 4).

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

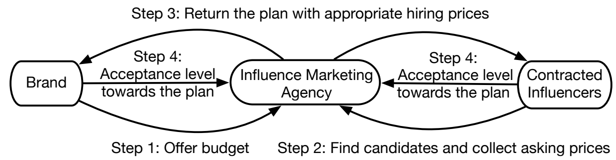


Figure 1: The business model of influencer marketing

Due to its importance and great usefulness, influencer marketing has received considerable attention in the field with different focus [30, 32, 55, 57, 60, 63]. While significant efforts have been paid to help the agency find suitable influencers for the brand (in Step 2) based on influencers' asking prices, existing studies on influencer selection barely consider whether the asking prices of the returned influencers as the final hiring prices are acceptable to the brand. To see this, let us take the Influence Maximization (IM) problem as an example. In the IM problem, the goal is to hire a *seed set* of candidate influencers within the given budget from the brand to maximize the overall influence. In this setting, the seed set is found based on influencers' asking prices which are assumed to be the final hiring prices and *unconditionally* accepted by the brand.

Unfortunately, we argue that such an assumption is barely true in real-world influencer marketing campaigns; instead, as we will elaborate next, in actual campaigns, there inevitably are conflicting interests – the brand wants a high return of investment whereas the influencers have high asking prices (i.e., expect high profits). That explains the nature of Step 3 in reconciling such conflicts, where the agency tries to rationalize the asking prices and reach an agreement that hopefully benefits all parties [5].

The Brand's Point of View. Instead of the abstract notion of influence, what the brand truly cares about is the *Return of Investment* (ROI), i.e., what the brand would get in return from their investment. As a result, in addition to the influence, the *engagement rate* of an influencer matters. Here, the engagement rate is the percentage of the influence (e.g., the number of followers) of an influencer that are actually effective (e.g., how many followers are engaged by the influencer) in a campaign. Therefore, the engagement rate reflects the actual return to the brand. Interestingly, according to recent studies [4, 8, 13, 14], influencers with a relatively small influence often have higher engagement rates than those with high influence. Taking Twitter as an example, the engagement rate of influencers with about 10,000 followers is generally 83% higher than that of the influencers with 100,000+ followers [8]. Therefore, from the brand's point of view, hiring 10 influencers each with 10,000 followers should be a better deal than spending the same cost on hiring just one influencer with 100,000 followers, simply because the former deal brings higher engagement rates of those influencers. In other words, a "big" influencer with great influence would *not* be competitive in the market if s/he has a *price rate* (i.e., price per unit of influence) more expensive than those "small" influencers with higher engagement rates [3, 5, 20]. For example, some UK brands were willing to pay for those influencers with no more than 10,000 followers at two times larger price rates than paying celebrities with ≥ 1M followers [3]. Therefore, for the brand, a campaign plan is ***investment-persuasive*** only if big influencers have *no more expensive* price rates than small ones.

The Influencers' Point of View. For the influencers, high profits are certainly the most important expectation. Therefore, their asking price rates tend to be close to the high end of their corresponding market price range. As such, the asking prices (which are visible to the agency *only*) are generally *not* investment-persuasive to the brand when used as the final hiring prices.

The Agency's Point of View. There is a clear conflict of interests between the brand and the influencers in the campaign cost. Here, the agency's job is to reconcile such a conflict to make the campaign a deal – the agency needs to decide *appropriate hiring prices* for the influencers such that the resulted campaign plan is investment-persuasive to the brand, and meanwhile satisfies the influencers' expectations with the best effort.

The Problem to be Solved. Motivated by the above, we play the role of the agency in campaigns and hope to reconcile the above conflict by rationalizing prices in Step 3 of Figure 1. Specifically, we formulate the agency's job as the problem of *Profit Divergence Minimization in Investment-Persuasive Campaign* (PDMIC). Given a brand's budget and a set of influencers' asking prices and influences, the goal of the PDMIC problem is to make an investment-persuasive campaign (i.e., a pricing scheme for the influencers) that can minimize the *profit divergence* of the influencers. Here, the profit divergence is defined as the sum of the *absolute difference* between the *actual hiring price* and the asking price of each influencer.

Example 1. *In Table 1, the left part lists five candidates, their influences and asking prices (visible to the agency only), and three campaigns. Given a total budget of $600, we aim to compute an* investment-persuasive *campaign. That is, influencers with the same influence charge the same price rate (i.e., price per unit of influence), and the price rate of an influencer with greater influence cannot be larger than that of an influencer with less influence. Here, both Campaign 1 and Campaign 2 are investment-persuasive. In Campaign 2, v_5 is not hired due to insufficient remaining budget. Campaign 1 achieves a much smaller total profit divergence (i.e., 60 vs. 130) and is obviously better because it makes the best use of the budget, helps achieve larger advertisement exposure for the brand and meanwhile satisfies the profit needs of more influencers with the best effort. On the other hand, Campaign 3 is not qualified since it is not investment-persuasive: the influencers v_2 to v_5 with the same influence are hired with different price rates. Apparently, Campaign 3 will not be accepted by the brand that emphasizes return on investment. In this case, the brand will expect their prices to be $70, which is the lowest hiring price among them but way below the asking prices of v_2, v_3 and v_4.*

Benefits of investment-persuasive campaigns. Aiming at investment-persuasive campaigns has a number of benefits. First, the investment-persuasive constraint makes a campaign "value for money" and hence benefits the brand, who cares about ROI rather than abstract notion of influence without considering realistic factors (e.g., engagement rate). Second, the objective tries to minimize the deviation of the actual hiring prices from the influencers' asking prices, which maintains a good balance between the interests of the brand and influencers in the hiring costs. Last but not least, it helps maintain an environment of *fair competition* in the market. Specifically, the aim of achieving investment-persuasive campaigns allows us to detect "speculators" who have much lower asking price rate than competitors, so as to avoid engrossing the market. Furthermore, this aim also prevents influencers, who have much higher asking price rates than competitors, from harming the benefit of the brand. For instance, in Table 1, the asking price rates (i.e., 1.3 and 0.7) of v_2 and v_5 may be unreasonable as they are notably different from that (i.e., 1) of the majority of their competitors v_1, v_3 and v_4. Thus, a good investment-persuasive campaign (e.g., Campaign 1) is able to adjust their actual price rates based on the overall market.

Table 1: Three campaign plans. Meaning of the acronyms: C. for Candidates, Inf. for Influence, Ask. for Asking price ($), P. for Hiring Price ($), R. for Hiring Price Rate, Dif. for Absolute Difference (i.e., |Ask.–P.|), and Camp. for Campaign.

C.	Inf.	Ask.	Camp. 1			Camp. 2			Camp. 3	
			P.	R.	Dif.	P.	R.	Dif.	P.	R.
v_1	200	200	200	1	0	200	1	0	200	1
v_2	100	130	100	1	30	130	1.3	0	130	1.3
v_3	100	100	100	1	0	130	1.3	30	100	1
v_4	100	100	100	1	0	130	1.3	30	100	1
v_5	100	70	100	1	30	0	0	70	70	0.7

The role of the PDMIC problem. We assume that the candidate influencers have been recommended already in the Step 2 of Figure 1 based on different possible goals (e.g., influence maximization [63] or regret minimization [30]), and the PDMIC problem focuses on rationalizing the prices of candidates in the Step 3. Thus, our problem serves a totally different role from existing candidate selection problems in the business model.

To summarize, we make the following contributions:

- To our best knowledge, we are the first to comprehensively study the vital role of the agency that has been oversimplified or overlooked in previous work. Specifically, we formulate the job of the agency as the problem of the Profit Divergence Minimization in Investment-Persuasive Campaigns (PDMIC), which aims to attract investments while protecting stakeholders' benefits and helping maintain an environment of fair competition. (Section 3)
- We prove the NP-hardness of the PDMIC problem. (Section 4)
- As to be discussed in subsequent sections, the challenge of the problem mainly lies in the extremely large searching space of the hiring prices for the influencers. As such, we address the problem by considering a small searching sub-space at first, and then gradually expanding the searching sub-space to the whole space in the end. In other words, we consider gradually relaxing the restriction on the price choices from strict ones to none (i.e., no restriction in the end). For solving the problem under these different settings, we propose effective and efficient algorithms:
  - As a first step, we consider the *binary-choice* restriction, where each influencer is required to be either hired with a specified price or unhired with no cost. Under such a restriction, we first propose an *exact* dynamic-programming based algorithm. And then, we show a 2-approximate algorithm which often produces high-quality solutions (i.e., with low profit divergence) in our experiments. (Section 5)

  – We propose an exact algorithm under a more relaxed restriction that the hiring price of each influencer could be chosen from a specified set of *integer* prices. (Section 6)
  – When no restriction is imposed, that is, we can select any price, two fast yet effective heuristic algorithms are carefully designed. (Section 7)
- We propose three interesting metrics for effectiveness evaluation, which might be useful to future efforts on this topic. Our extensive experiments on real-world datasets show that our ultimate approximate methods (with no restriction): (i) notably outperform the binary-choice based methods in terms of result quality, and (ii) can even achieve results with better quality than our exact integer-price-choices algorithm with a sufficiently large number of integer price choices, and meanwhile achieve up to eight-orders-of-magnitude speedup. (Section **??**)

2 RELATED WORK

In this section, we give a literature review on influencer marketing and resource allocation. The literature on influencer marketing can be divided into two classes: (i) influencer selection optimization and (ii) pricing scheme optimization. Studies of the first class only focus on Step 2 in Figure 1, that is selecting influencers based on their asking prices for specific objectives (e.g., influence maximization). Moreover, they never consider whether the asking prices of selected influencers are acceptable or reasonable to the brand. In constrast, we focus on Step 3 in Figure 1 and study rationalizing the prices of the selected influencers. Studies of the second class are also drastically different from our study, as they either have *different* pricing focus (e.g., products) or *do not* consider the conflicts of stakeholders' interest. Our PDMIC problem is essentially a useful downstream application of the popular influencer selection optimization problem and the pricing scheme optimisation problem in the field. PDMIC is orthogonal to those problems studied in this line of research, yet the input of PDMIC can be assumed to be a certain output from influencer selection algorithms as a pre-processing.

Influencer Selection Optimization. *Influence maximization* aims to select a limited number of influencers with the greatest influence spread. There are considerable studies with the goal of improving the efficiency of influence estimation under different stochastic diffusion models and improving the influencer selection [27, 31, 33, 35, 39–41, 43, 44, 48, 48, 51, 52, 54, 58, 59, 61–63, 67–70, 76, 78, 79, 84, 85, 87, 89–91]. It provides deep insights on collective behavior of users in online social network and is of great importance in understanding influence cascades in viral marketing. This inspires many subsequent works on different variations. *Budgeted influence maximization* [32, 56, 68, 77, 80] considers different prices for hiring influencers and aims to select a set of influencers with the greatest influence under limited budgets. *Profit maximization* [57, 88] aims to choose a set of influencers who are able to bring the maximum profit which is calculated by influencers' influence spread minus the costs of hiring them. *Revenue maximization* [29, 55] extends budgeted influence maximization by maximizing revenue proportional to influence and considering finding different disjoint seed sets for different advertisements. *Regret minimization* [30, 98] aims to select a set of influencers to minimize difference between the revenue proportional to the influence brought by the selected influencers and the budget of the brand.

Pricing Scheme Optimization. Zhu et al. [99] study the problem of pricing influencers with a totally different objective which aims to make their prices effectively reflect their unique contributions to the influence of different seed sets. Therefore, the computed campaign is not investment-persuasive. In particular, influencers with the same influence can be assigned with very different hiring prices and influencers with the greater influence can charge much higher price rates. Arthur et al. [28] aim to decide the least discounts of a product to encourage consumption and expect the buyer to further propagate the product information to other online users. Chen et al. [42] study how changes of the network structure affect the discriminatory pricing strategies where companies sell products with discounts to users with large centrality or influence. Outside the domain of influencer marketing, there are also many studies on pricing strategies but with different contexts and targets (e.g., query [46, 66], data [36, 72], solutions [38] and crowdsourcing [92]).

Resource Allocation Optimization. It is a very broad line of research which aims to compute the optimal distributions of resources among competing alternatives in order to maximize the objective score [21], where the definitions of resources and alternatives depend on specific problem contexts. For instance, they refer to the knapsack space and items, the budget and influencers, and the budget and locations for deploying facilities in variants of the knapsack problem [34, 65, 74, 81, 82, 86], the influence maximization problem [29, 30, 32, 55, 56, 63, 68, 98] and the location selection problem [71, 75, 83, 93, 94, 96, 97], respectively. With different contexts, constraints, and objectives, existing studies on these problems face different challenges and effective solutions require problem-specific designs. Thus, it is unrealistic to have a single general and simple algorithm which fits all problems above. Let us take the greedy strategy, which iteratively selects an alternative with the greatest marginal gain to the objective per unit of consumed resources, for an example. This strategy produces approximate solutions for the classical influence maximization problem [63] but does not have any theoretical guarantee for the budgeted influence maximization or the classical knapsack problem. In the latter two problems, extensions of this greedy strategy are required to have approximation ratios [1, 32]. However, extensions may not always be an effective remedy especially in problems with more problem-specific objectives and constraints. For instance, in the billboard placement problem [97] where the objective is not submodular, a specific branch-and-bound method was proposed to solve the problem exactly since the greedy strategy with any possible extension will not produce high-quality solutions theoretically.

Therefore, a simple yet general resource allocation strategy with possible extensions is unlikely to be highly effective in our problem which has a unique investment-persuasive constraint. To our best knowledge, there is no resource allocation study that considers such a constraint. However, for a better connection to existing work in this field, we propose two greedy based solutions with different selection criteria in Section 5.2 and Section 7.1 respectively. Despite that these two extensions achieve promising results in some cases, they are still notably outperformed by our advanced method with more problem-specific design (Section 7.2).

3 PROBLEM FORMULATION

In this paper, we focus on the job of the agency in Step 3 of Figure 1, that is, the step to design a plan specifying *which* influencers to be hired at *what* costs with respect to the given *budget B*. Next, we formulate this problem and frequent notations are in Table 2.

Budget. A *budget*, denoted by B, is the overall available cost for hiring influencers in the campaign.

Influencer. An influencer v is a candidate to be hired in the campaign, who is associated with the following three attributes.

Influence. Each influencer v is associated with a *positive real number* representing her *influence*, denoted by $\delta(v)$, which is measured under certain *influence measurement*. However, the choice of the influence measurement is *irrelevant* to our problem formulation and the computation of $\delta(v)$ is *orthogonal* to our problem. And the choice of the influence measurement is often up to the context of application scenarios. For example, $\delta(v)$ can be as simple as *the number of followers* of v or can be the popular influence notions in the context of *Influence Maximization* defined under the independent cascade model [50] and the linear threshold model [53].

Asking Price. Each influencer v specifies an *asking price* p^v_{ask} visible to the agency only. The asking price of v is often represented by a *percentage* of the budget known to the agency only, e.g., 5% of B. As a result, for the ease of discussion, we simply consider p^v_{ask} as the *normalized price with respect to the budget B* that v wants to be hired with. In other words, the sum of the asking prices over all the influencers is *exactly* equal to B.

Acceptable Hiring Price Threshold. In addition to the asking price, each influencer v also specifies a non-negative *acceptable hiring price threshold*, denoted by $\tau(v) \geq 0$, which is the hiring price threshold that would be acceptable to v. That means, influencer v will not take the job if the offered price is *smaller* than $\tau(v)$. In particular, $\tau(v) = 0$ indicates influencer v does not care about the hiring price as long as it is *positive* (i.e., > 0). However, v will never take the job if the offered hiring price is 0: an influencer will never work for free. Therefore, being offered a 0 hiring price is equivalent to the case that the corresponding influencer is *not* hired.

Campaign. Given a budget B and a set F of influencers, i.e., $F = \{v_1, v_2, \ldots, v_{|F|}\}$, a *campaign* is a *pricing scheme*, denoted by $\mathcal{P} = [\mathcal{P}(v_1), \mathcal{P}(v_2), \ldots, \mathcal{P}(v_{|F|})]$, which assigns a *hiring price* $\mathcal{P}(v)$, to each influencer $v \in F$ such that: (i) $\mathcal{P}(v) \geq \tau(v)$ if v is hired to endorse the campaign, or $\mathcal{P}(v) = 0$ if v is not hired, and (ii) $\sum_{v \in F} \mathcal{P}(v) \leq B$. For the ease of presentation, we directly use h as v_h to refer to the h^{th} influencer in F.

Definition 1 (Profit Divergence of a Campaign). *Consider a campaign $\mathcal{P}$ with respect to budget B and influencer set F; the* profit divergence *of $\mathcal{P}$, denoted by $\mathcal{D}_{\mathcal{P}}$, is the total absolute difference between the hiring price and the asking price of each candidate, that is, $\mathcal{D}_{\mathcal{P}} = \sum_{v \in F} |\mathcal{P}(v) - p^v_{ask}|$.*

Definition 2 (Investment-Persuasive Campaign). *A campaign $\mathcal{P}$ is* investment-persuasive *if the hired influencers with higher influence have lower* hiring price rates*, i.e., the hiring price per unit of influence. Formally, $\forall u, v \in F$, if both u and v are hired, i.e., $\mathcal{P}(v) \neq 0$ and $\mathcal{P}(u) \neq 0$, and u has no greater influence than v does, i.e., $\delta(u) \leq \delta(v)$, then $\frac{\mathcal{P}(v)}{\delta(v)} \leq \frac{\mathcal{P}(u)}{\delta(u)}$ holds.*

To adjust the persuasiveness level of the campaign, we can also control the scale of difference among the hiring price rates of influencers with different influences. For example, in the definition above, we can specify that the hiring price rate of v must be at least twice smaller than that of u. Here, we allow the existence of the equal condition for the ease of presentation, since our theoretical analysis and methods are orthogonal to the scale of difference.

Definition 3 (Profit Divergence Minimization in Investment-persuasive Campaigns (PDMIC)). *Given a budget B and a set F of influencers, the goal of PDMIC is to return an investment-persuasive campaign $\mathcal{P}^*$ with the minimum profit divergence. Mathematically, PDMIC can be formulated as the following the problem:*

$$\mathcal{P}^* = \arg\min_{\mathcal{P}} \mathcal{D}_{\mathcal{P}}$$

subject to:

- *(i)* normalized asking prices: $\sum_{v \in F} p^v_{ask} = B$,
- *(ii)* non-negative hiring prices: $\mathcal{P}(v) \geq 0, \forall v \in F$,
- *(iii)* acceptable hiring price thresholds: $\mathcal{P}(v) \geq \tau(v), \forall \mathcal{P}(v) > 0$,
- *(iv)* within the budget: $\sum_{v \in F} \mathcal{P}(v) \leq B$, *and*
- *(v)* being investment-persuasive: $\frac{\mathcal{P}(v)}{\delta(v)} \leq \frac{\mathcal{P}(u)}{\delta(u)}$, *for all* $\mathcal{P}(v) \cdot \mathcal{P}(u) > 0 \wedge \delta(u) \leq \delta(v)$.

Application Scenarios beyond Influencer Marketing. In fact, our investment-persuasive constraint is an implementation of the tiered pricing model, where prices are offered with different levels of service or quality [15]. In the tiered pricing model, a higher-tiered service charges more yet its price rate (i.e., price per value) needs to be at least as good as a lower-tiered service [22]. Our investment-persuasive constraint nicely captures this property. As a result, our PDMIC problem can be directly applied to various pricing scenarios where the tiered pricing model is considered. Below, we just name two application scenarios.

Scenario 1: Cloud Service. For instance, when a company wants to adopt cloud services (e.g., multi-cloud or hybrid cloud services [9, 23] sourced from different vendors) in the cloud computing business, an SaaS broker is usually involved as a middle man between the company and a set F of service providers to negotiate a Service Level Agreement (SLA) [47, 95]. In this case, we can consider that the buyer has a budget B and each service provider $v \in F$ has an asking price p^v_{ask}, an acceptable hiring threshold $\tau(v)$ and a value $\delta(v)$ (e.g., storage space and processing speed). To protect the benefits of the buyer and providers while attracting investments from the buyer, the broker may want to compute the investment-persuasive campaign with the minimum profit divergence.

Scenario 2: Material Supply. Similarly, when a company seeks for certain materials (e.g., carbon fiber, plastic and residue waste), a material brokerage company [2, 10] often plays an important role to achieve a win-win solution (i.e., an investment-persuasive campaign) between the buyer and the sellers. In general, one single seller may not be able to satisfy the need of the buyer company. As a result, such a solution often contains multiple sellers which have limited supply of the material with different qualities $\delta(\cdot)$.

Table 2: Frequently used notations

Notation	Description
B	The campaign budget.
$\delta(v)$	The influence of influencer v.
p^v_{ask}	The asking price of v.
$p^v_{ask}/\delta(v)$	The asking price rate of v.
$\tau(v)$	The acceptable hiring price threshold of v.
$[\tau(v), B]$	The acceptable hiring price range of v.
$\mathcal{P}(v)$	The hiring price of v in a campaign $\mathcal{P}$.
p_v	A candidate price for hiring v.

4 HARDNESS AND SOLUTION OVERVIEW

In this section, we first show the NP-hardness of our PDMIC (Definition 3) and then give an overview of our solutions. More specifically, our NP-hardness proof is a reduction from the Knapsack Problem [74] to a special input instance of our PDMIC, where for each influencer v, we have $\delta(v) = \tau(v)$ and $p^v_{ask} \le \delta(v)$.

Lemma 1. *For the input instance satisfying: for each influencer v, $\delta(v) = \tau(v)$ and $p^v_{ask} \le \delta(v)$, the optimal solution $\mathcal{P}^*$ to PDMIC satisfies: $\mathcal{P}^*(v) = \delta(v)$ if v is hired, and $\mathcal{P}^*(v) = 0$ otherwise.*

Proof. Let F^* to be set of influencers that are hired by the optimal solution $\mathcal{P}^*$ to this special input instance.

First, according to Constraint (iii) in Definition 3, we know that $\mathcal{P}^*(v) \ge \tau(v)$ for all $v \in F^*$. Since $\delta(v) = \tau(v)$ and $p^v_{ask} \le \delta(v)$, we have $\mathcal{P}^*(v) \ge \tau(v) = \delta(v) \ge p^v_{ask}$ for all $v \in F^*$.

Second, observe that any pricing scheme $\mathcal{P}$ with $\mathcal{P}(v) = \delta(v)$ for all hired influencers v satisfies the investment-persuasive constraint (i.e., Constraint (v) in Definition 3); this is because in this case, every hired influencer has price rate of 1. Moreover, by the definition of profit divergence (Definition 1) and since $\mathcal{P}^*(v) \ge \delta(v) \ge p^v_{ask}$ for all $v \in F^*$, decreasing the hiring price $\mathcal{P}^*(v)$ for each $v \in F^*$ (subject to Constraint (v)) can decrease the profit divergence $\mathcal{D}_{\mathcal{P}^*}$. Therefore, $\mathcal{P}^*(v) = \delta(v)$ must hold for each $v \in F^*$. □

According to Lemma 1, the optimal PDMIC solution to the aforementioned special input instances satisfying $\mathcal{P}^*(v) = \delta(v)$ *if and only if* v is hired. For ease of reference, we formulate a special case of PDMIC as below and refer it as *Binary PDMIC*. Note that the Binary PDMIC is slightly more general than the special input instance described in Lemma 1. Specifically, we allow $\delta(v) \ge \tau(v)$ and the constraint $p^v_{ask} \le \delta(v)$ does not necessarily exist.

Definition 4 (Binary PDMIC). *Given a budget B, a candidate set F, the goal is to hire influencers from F under budget B with minimum divergence score:*

$$\text{minimize} \sum_{h=1}^{|F|} |x_h \cdot p_h - p^h_{ask}|, \qquad (1)$$

subject to: (i) $\sum_{h \in F} p^h_{ask} = B$, (ii) $x_h \in \{0, 1\}$, (iii) $p_h = \delta(h) \ge \tau(v)$, and (iv) $\sum_{h=1}^{|F|} x_h \cdot p_h \le B$.

Define function $f(h, p_h) = p_h$ if $p^h_{ask} > p_h$, and $f(h, p_h) = 2p^h_{ask} - p_h$ otherwise. Taking off the absolute operations, Objective (1) is equivalent to minimize $B - \sum_{h=1}^{|F|} x_h \cdot f(h, p_h)$, and hence, further equivalent to:

$$\text{maximize} \sum_{h=1}^{|F|} x_h \cdot f(h, p_h) . \qquad (2)$$

Next, we show that Binary PDMIC is NP-hard by a reduction from the Knapsack Problem [74]. Specially, we will first prove the NP-hardness of a variation of the Knapsack Problem, called *Knapsack Problems with Value and Capacity Constraints* (KPVC) (i.e., Definition 6 as below). Then we perform a reduction from KPVC to the Binary PDMIC.

Definition 5 (Knapsack problem (KP) [74]). *Given a set of i items, each of which has a non-negative weight w_h and a value z_h ($1 \le h \le i$), and a bag with capacity $\bar{B}$, the knapsack problem aims to choose a subset of items into the bag without exceeding capacity $\bar{B}$ while maximizing the total value, i.e. maximize $\sum_{h=1}^{i} x_h \cdot z_h$, subject to (i) $x_h = 1$ or 0 and (ii) $\sum_{h=1}^{i} x_h \cdot w_h \le \bar{B}$.*

Definition 6 (Knapsack problem with value and capacity constraint (KPVC)). *The KPVC problem is a special variant of the KP problem with two extra constraints on the input: (i)* Value Constraint*: for each item h ($1 \le h \le i$), its value is at most its weight, i.e., $z'_h \le w'_h$, and (ii)* Capacity Constraint*: the capacity of the bag satisfies $\bar{B}' = (\sum_{h=1}^{i} w'_h + z'_h)/2$.*

Lemma 2. *The KPVC problem is NP-hard.*

Proof. We prove the NP-hardness via reduction from the input instance for the KP problem to a valid input for the KPVC problem.

Meeting the Value Constraint. Our first step is to modify the above KP input instance to satisfy the Value Constraint. The crucial idea is that the KP optimization does not change, when both the item weights and the bag capacity are scaled by a same *positive* factor. Hence, let $t = \max_{h=1}^{i} z_h/w_h > 0$. We scale the item weights w_h to $w'_h = w_h \cdot t$ for $1 \le h \le i$, and the bag capacity $\bar{B}$ to $\bar{B}' = \bar{B} \cdot t$. It can be verified that $\sum_h w_h \le \bar{B} \Leftrightarrow \sum_h w'_h \le \bar{B}'$. Furthermore, by setting $z'_h = z_h$, we have $z'_h = w_h \cdot \frac{z_h}{w_h} \le w_h \cdot t \le w'_h$ for all h's.

Meeting the Capacity Constraint. Based on the scaled input instance above, we further strengthen it to meet the Capacity Constraint ($\bar{B}' = (\sum_{h=1} w'_h + z'_h)/2$). There are two possible cases.

Case 1: $\bar{B}' > (\sum_{h=1} w'_h + z'_h)/2$. We introduce a "dummy" item $(i+1)$ with weight and values as $w'_{i+1} = 2\bar{B}' - (\sum_{h=1}^{i} w'_h + z'_h)$ and $z'_{i+1} = 0$ respectively. This "dummy" item will never be chosen since its value is 0. As all other items and the capacity remains the same, an optimal solution to the this KPVC instance is also an optimal solution to the original KP instance.

Case 2: $\bar{B}' < (\sum_{h=1} w'_h + z'_h)/2$. We introduce a "powerful" item $(i+1)$ with value and weight as $z'_{i+1} = 1 + \sum_{h=1}^{i} z'_h$ and $w'_{i+1} = z'_{i+1} + \sum_{h=1}^{i}(w'_h + z'_h) - 2\bar{B}' > z'_{i+1}$ respectively. Moreover, we set the capacity $\bar{B}^* = \bar{B}' + w'_{i+1}$. It can be verified that this super useful item must be in the optimal solution for the KPVC input instance under $\bar{B}^*$. This is because the value of the super useful item is greater than the value sum of all other items. Excluding this "powerful" item in the optimal solution of the KPVC input instance also gives an optimal solution to the original KP input.

Clearly, the above reduction can be perform in polynomial time. The KPVC problem is thus NP-hard. □

Table 3: Overview of our proposed solutions

Hiring Price Choices	Method	Solution	Time Complexity
Binary prices (Sec 5)	BC-Exact BC-MG	Exact Approx.	$O(\lvert F\rvert \cdot B)$ $O(\lvert F\rvert \log \lvert F\rvert)$
A set of integer prices (Sec 6)	IC-Exact	Exact	$O(\lvert F\rvert^2 \lvert R_{\max}\rvert^2 B)$
Any price above the price threshold (Sec 7)	CR-Inf CR-MWS	Heuristic	$O(\lvert F\rvert \log \lvert F\rvert)$ $O(\lvert F\rvert^2)$

Theorem 1. *The Binary PDMIC (Definition 4) problem is NP-hard.*

Proof. We reduce KPVC to the Binary PDMIC, by mapping $|F| = i$, $p^h_{ask} = (w'_h + z'_h)/2$, $B = \bar{B}'$, and $\delta(h) = \tau(h) = p_h = w'_h$. By the Value and Capacity constraint in KPVC, we have $p^h_{ask} \leq p_h = \delta(h)$ and $\sum_{h=1}^{|F|} p^h_{ask} = B$. Based on Expressions (1) and (2), the objective becomes: maximize $\sum_{h=1}^{|F|} x_h(2p^h_{ask} - p_h) = \sum_{h=1}^{i} x_h \cdot z'_h$, which is equivalent to the objective of KPVC. The optimal solution of this instance of Binary PDMIC implies an optimal solution to the KPVC input instance. Since the above reduction can be performed in polynomial time, the Binary PDMIC problem is NP-hard. □

By Lemma 1 and Definition 4, we know that Binary PDMIC is a special case of PDMIC. Therefore, the corollary below follows immediately from Theorem 1.

Corollary 2. *The PDMIC problem (Definition 3) is NP-hard.*

Solution overview. Considering the potentially extreme large search space of the continuous acceptable price range $[\tau(\cdot), B]$, we tackle the PDMIC problem by incrementally relaxing the restrictions on price choices (i.e., from the smallest sub-space to a larger sub-space and then to the whole space in the end). Specifically, we start from the binary-price-choices restriction where each influencer can only hired with a fixed price or not hired with zero cost, and develop the *Binary-Choice based Exact method* (BC-Exact) and *Binary-Choice and Minimum Gain based method* (BC-MG). Then, we make further relaxation by allowing the price to be selected from a set of integer choices, and propose the *Integer-Choice based Exact Method* (IC-Exact). Finally, we allow the price to be selected in the continuous acceptable price range (i.e., the whole space), and propose *Continuous-Range and Influence based Method* (CR-Inf) and *Continuous-Range and Maximum-Weighted-Subsequence based Method* (CR-MWS). Table 3 summarizes our methods. Note that all the theoretical claims on any solution's effectiveness in this paper are based on the *price choices* built from the price range $[\tau(\cdot), B]$.

5 PDMIC WITH BINARY PRICE CHOICES

In this section, we focus on solving the Binary PDMIC problem (defined in Definition 4). By Theorem 1, we know that this problem is NP-hard. Moreover, the proof of Theorem 1 indeed shows a subtle connection between the Binary PDMIC problem and the Knapsack problem. Motivated by this, we first adopt the dynamic programming algorithm for the latter problem to find optimal solutions for the Binary PDMIC. To improve the efficiency, we propose a greedy method that can produce competitive solutions with a 2-approximate guarantee while achieving significant speedups.

Algorithm 1: BC-Exact

Input : A set F where each candidate v can only be hired with $p_v = \delta(v)$ and has an asking price p^v_{ask}, and $B = \sum_{v \in F} p^v_{ask}$.
Output: Divergence score.

Initialize all entries M of size $(|F| + 1) \cdot (B + 1)$ with 0;
for $h = 1$ *to* $|F|$ **do**
  for $b = 1$ *to* B **do**
    if $p_h \leq b$ **then**
      $score = f(h, p_h) + M[h-1][b-p_h]$;
      $M[h][b] = \max(score, M[h-1][b])$;
    else $M[h][b] = M[h-1][b]$;
Initialize the campaign $\mathcal{P}$: $\forall v \in F$, $\mathcal{P}(v) = 0$;
Compute $\mathcal{P}$ by backtracking from $M[|F|][B]$;
return $B - M[|F|][B]$;

5.1 An Exact Method

In this subsection, we present a *Binary-Choice based Exact method* called BC-Exact. The basic idea of BC-Exact is to perform dynamic programming to maintain a matrix M of size $(|F| + 1) \times (B + 1)$, where each entry $M[h][b]$ (for $1 \leq h \leq |F|, 0 \leq b \leq B$) stores the optimal value of Objective (2) with respect to a budget b and considering the first h influencers only. With the budget limit b, the optimal selections among the first h influencers can only result from two possibilities depending on whether influencer h is hired or not. Specifically, if influencer h is *not* hired, then $M[h][b] = M[h-1][b]$. Otherwise, if h is hired, $M[h][b]$ is equal to $f(h, p_h)$ plus the maximum value obtained by considering the first $h - 1$ influencers with budget limit $b - p_h$, i.e., $M[h][b] = f(h, p_h) + M[h-1][b-p_h]$. Therefore, $M[h][b]$ can be expressed as the greater objective value between these two cases, namely,

$$M[h][b] = \max\{M[h-1][b], M[h-1][b-p_h] + f(h, p_h)\}.$$

Each entry $M[h][b]$ can be computed by a simple recursion. Once $M[|F|][B]$ is computed, BC-Exact returns $B - M[|F|][B]$ as the minimum divergence.

Note that, if we set the price equal to the influence, the influence must be an integer to make this approach feasible because the computation of the index of the second dimension of M is based on prices. If we use other influence measurements (e.g., influence spread under the independent cascade model [50]), we can ignore the fractional part because it is significantly smaller than the integer part especially for influencers.

Running Time Analysis. While the idea of this dynamic programming approach is simple, it requires a *pseudo*-polynomial time complexity $O(|F| \cdot B)$ to compute M, which is indeed not polynomial in the problem input size and hence not scalable to large budgets.

5.2 A Minimum Gain based Method

To achieve higher scalability, we propose a *Binary-Choice and Minimum Gain based method* (called BC-MG) which can produce 2-*approximate* solutions in only $O(|F| \log |F|)$ time.

Before describing the algorithm, we first rewrite Objective (1). Let $\gamma = B - \sum_{h=1}^{|F|} f(h, p_h)$ and $X = \sum_{h=1}^{|F|} f(h, p_h) - \sum_{h=1}^{|F|} x_h \cdot f(h, p_h)$. Then Objective (1) can be rewritten as:

$$\text{minimize } \gamma + X \tag{3}$$

The basic idea of BC-MG is to approximate the optimal value of $\gamma + X$, denoted as $OPT_{\gamma+X}$, with an approximation of the optimal

Algorithm 2: BC-MG

Input : A set F where each candidate v has an asking price p^v_{ask} and can only be hired with $p_v = \delta(v)$, and $B = \sum_{v \in F} p^v_{ask}$.
Output : Divergence score.

1 $S_c = \emptyset$, $B_c = \sum_{v \in F} p_v - B$, $f_{min} = B$;
2 Sort $v \in F$ based on $f(v, p_v)/p_v$ in non-decreasing order.;
3 **foreach** $v \in F$ **do**
4   **if** $p_v > B_c$ **then**
5   **if** $f(v, p_v) < f_{min}$ **then** $v_{min} = v$; $f_{min} = f(v, p_v)$;
6   **else**
7   **if** $f(v, p_v)/p_v \leq f_{min}/B_c$ **then**
8             $S_c = S_c \cup \{v\}$; $B_c = B_c - p_v$;
9   **if** $B_c = 0$ **then** Break;
10   **else** $S_c = S_c \cup \{v_{min}\}$; $B_c = 0$; Break ;
11 **if** $B_c > 0$ **then** $S_c = S_c \cup \{v_{min}\}$;
12 Initialize the campaign $\mathcal{P}$: $\forall v \in F$, $\mathcal{P}(v) = 0$;
13 **foreach** $v \in F \setminus S_c$ **do** $\mathcal{P}(v) = p_v$;
14 $\gamma = B - \sum_{v \in F} f(v, p_v)$;
15 **return** $\gamma + \sum_{v \in S_c} f(v, p_v)$;

value OPT_X of X. Observe that minimizing X is equivalent to finding a subset $S^* \subseteq F$, whose accumulated $f()$ score (i.e., the second term in the expression of X) is maximized under budget B. Moreover, it can be verified that the value of X is essentially the accumulated $f()$ score of the influencers in the *complement set* $S^*_c = F \setminus S^*$ of S^*. As a result, minimizing X is equivalent to $\min_{S_c \subseteq F} \sum_{v \in S_c} f(v, p_v)$, subject to $\sum_{v \in S^*} p_v = \sum_{v \in F} p_v - \sum_{v \in S_c} p_v \leq B$, and hence, equivalently, $\sum_{v \in S_c} p_v \geq \sum_{v \in F} p_v - B$.

Let B_c be the current available budget; initially, $B_c = \sum_{v \in F} p_v - B$. The basic idea of BC-MG is to iteratively add an influencer u with the minimum gain per unit price (i.e., $f(u, p_u)/\min\{B_c, p_u\}$) into S_c until the current budget B_c is *exhausted* (i.e., $B_c \leq 0$). A straightforward approach is to, in each iteration, scan the current set of remaining influencers, i.e., $F \setminus S_c$, and select the influencer u with smallest $f(u, p_u)/\min\{B_c, p_u\}$ into S_c. However, this naive approach incurs $O(|F|^2)$ time complexity.

Here, we propose a more efficient approach, BC-MG (the pseudo code is shown in Algorithm 2), which runs in $O(|F| \log |F|)$ time. The first step of BC-MG is to sort $v \in F$ based on $f(v, p_v)/p_v$ in a non-decreasing order. Then BC-MG iteratively selects influencers into S_c according to this order. While this sorted list of the influencers may change with B_c decreasing, a crucial observation is that the rank of an influencer v will not change until the remaining budget $B_c < p_v$; and since then, the rank of v is calculated based on $f(v, p_v)/B_c$ rather than $f(v, p_v)/p_v$.

Thus, BC-MG maintains two pieces of information: (i) the sorted list of influencers with $p_v \leq B_c$, and (ii) the influencer v_{min} with the smallest $f()$ score among those with $p_v > B_c$. In each iteration, BC-MG just needs to compare the currently visited influencer v in the sorted list and v_{min}. If the gain per unit price of v is smaller than that of v_{min}, then add v to S_c and update $B_c \leftarrow B_c - p_v$ (i.e., Line 8 in Algorithm 2), and maintain v_{min} with respect to the updated B_c. Otherwise, add v_{min} to S_c and terminate the algorithm.

Running Time Analysis. Clearly, the time cost of each iteration is just $O(1)$, and there can be at most $|F|$ iterations. Thus, the time cost after sorting is bounded by $O(|F|)$. Putting together with the sorting cost of $O(|F| \log |F|)$, the running time is bounded by $O(|F| \log |F|)$.

A 2-Approximation Guarantee. We prove the theoretical guarantee (i.e., Theorem 3) with Lemma 3 below:

Lemma 3. *Let S_c denote the influencer set chosen by BC-MG with budget $B_c = \sum_{v \in F} p_v - B$. We have $\sum_{v \in S_c} f(v, p_v) \leq 2OPT_X$.*

Proof. Suppose all influencers are sorted based on their gain per unit price in non-decreasing order, instance I is our target instance where the budget $B_c = \sum_{v \in F} p_v - B$, S_c is the influencer set of size m found by BC-MG and OPT_X is the optimal value of X.

Let S^{m-1}_c denote the set of the first $m-1$ influencers in S_c, and the budget B^{m-1}_c be the total price of the first $m-1$ influencers. The accumulated $f()$ score of influencers in S^{m-1}_c must be minimum under budget B^{m-1}_c since the total price is exactly B^{m-1}_c and influencers are picked based on minimum gain per unit price. Replacing any influencer in S^{m-1}_c with one in $F \setminus S^{m-1}_c$ will not make the solution better. Observe that the optimal X value under a smaller budget cannot be larger and $B^{m-1}_c \leq B_c$. Thus, the accumulated $f()$ score of influencers in S^{m-1}_c is at most OPT_X.

Let $S^{last}_c = S_c \backslash S^{m-1}_c$ denote the set containing the last influencer in S_c. Suppose we have an instance I', where the $f()$ score of each first $m-1$ influencer is 0 and the budget is B_c. In instance I', the optimal solution set must consist of the first $m-1$ influencers and the influencer in S^{last}_c. It is obvious that the first $m-1$ influencers must be included as their $f()$ scores are 0. With the remaining budget, the optimal influencer selection is consistent with BC-MG and thus, the S^{last}_c must be included in the optimal solution of instance I'. Since the accumulated $f()$ score of the optimal solution in instance I' is clearly a lowerbound of OPT_X and S^{last}_c must be a part of the optimal solution, the $f()$ score from S^{last}_c cannot be greater than OPT_X. Thus, the accumulated $f()$ score of influencers from $S^{m-1}_c \cup S^{last}_c = S_c$ is not greater than $2 \cdot OPT_X$. □

Theorem 3. *Let $OPT_{\gamma+X}$ denote the optimal value of Objective* (3)*, and S_c denote the influencer set chosen by BC-MG under budget $B_c = \sum_{v \in F} p_v - B$. S_c is a 2-approximate solution such that $\gamma + \sum_{v \in S_c} f(v, p_v) \leq 2 \cdot OPT_{\gamma+X}$.*

Proof. Based on Lemma 3, we have $\sum_{v \in S_c} f(v, p_v) \leq 2 \cdot OPT_X$. Since $OPT_{\gamma+X} = \gamma + OPT_X$, we have $\gamma + \sum_{v \in S_c} f(v, p_v) \leq 2 \cdot OPT_X + \gamma \leq 2 \cdot (\gamma + OPT_X) = 2 \cdot OPT_{\gamma+X}$. Thus, Theorem 3 is proven. □

6 PDMIC WITH INTEGER PRICE CHOICES

As the binary-choice restriction might be too strict, the practical result quality of the aforementioned binary-choice algorithms may not be good enough. In this section, we relax this restriction by enlarging the search space – the price for hiring an influencer v can be 0 or chosen from a given set R_v of integers in the corresponding acceptable price range $[\tau(v), B]$. For this setting, we propose an exact algorithm, namely *Integer-Choice based Exact Method* (IC-Exact), which is also based on dynamic programming.

The pseudo code of IC-Exact is presented in Algorithm 3. Here, the price of each influencer $v \in F$ can be chosen from the set R_v, and we assume that elements in R_v are sorted in ascending order for ease of presentation. At the beginning, the influencers are sorted by their influence $\delta(v)$ in a *non-decreasing* order. The whole idea of this approach is to compute a matrix M of three dimensions. Next, we will describe the purpose of M and then how to compute M.

Algorithm 3: IC-Exact

Input : A set F of candidates where each candidate v has an asking price p^v_{ask} and an integer price choice set R_v, and $B = \sum_{v \in F} p^v_{ask}$.
Output : Divergence score.

1 Sort each $v \in F$ based on $\delta(v)$ in non-decreasing order;
2 Initialize entries of M with 0 where $M[h]$ has dims. $B \times |R_h|$;
3 **for** $b = 0$ *to* B **do**
4   **foreach** $p_1 \in R_1$ **do**
5   **if** $p_1 \leq b$ **then** $M[1][b][p_1] = f(1, p_1)$;
6 **for** $h = 2$ *to* $|F|$ **do**
7   **foreach** $p_h \in R_h$ **do**
8   **for** $b = p_h$ *to* B **do**
9             $M[h][b][p_h] = f(h, p_h)$;
10   **for** $j = 1$ *to* h*-1* **do**
11   **if** $\delta(j) = \delta(h)$ *and* $p_h \in R_j$ *and* $b - p_h \geq p_h$ **then**
12   // j must be hired with p_h to meet the investment persuasive constraint, and p_h is a feasible price for j
13                     $M[h][b][p_h] = \max(M[h][b][p_h], M[j][b - p_h][p_h] + f(h, p_h))$;
14   Continue;
15   **foreach** $p_j \in R_j$ **do**
16   **if** $b - p_h < p_j$ **then** Break;
17   **if** $p_j/\delta(j) \geq p_h/\delta(h)$ **then**
18                         $M[h][b][p_h] = \max(M[h][b][p_h], M[j][b - p_j][p_j] + f(h, p_h))$;
19 $k, p = \arg\max_{h \in [1,|I|], p_h \in R_h} M[h][B][p_h]$;
20 Initialize the campaign $\mathcal{P}$: $\forall v \in F$, $\mathcal{P}(v) = 0$;
21 Compute $\mathcal{P}$ by backtracking from $M[k][B][p]$;
22 **return** $B - M[k][B][p]$;

Purpose of matrix M. Each entry $M[h][b][p_h]$ records the $f()$ score sum of the optimal solution when we only consider hiring (i) the first h influencers (in the aforementioned sorted list) and (ii) the h-th influencer with price $p_h \in R_h$ under budget b. Clearly, the global optimal solution satisfies these two considerations with some h, p_h and b. Considering that increasing b while fixing h and p_h will not decrease the $f()$ score sum, the global optimal solution can be found at the largest entry among $M[\cdot][B][\cdot]$.

Computation of matrix M. If $p_h \geq b$, the h-th influencer cannot be hired and thus $M[h][b][p_h]$ is set as 0. Otherwise, $M[h][b][p_h]$ is initialized as $f(h, p_h)$ (Lines 5 and 9). It indicates that currently only influencer h is hired. Thus, we need to find the optimal solution when we only consider hiring the first $h-1$ influencers with the remaining budget $b - p_h$. Specifically, we need to find the entry $M[j^*][b - p_h][p_{j^*}]$ with the greatest value from candidate entries where each candidate entry $M[j][b - p_h][p_j]$ satisfies two requirements: (i) $j < h$ and $p_j \leq b - p_h$, and (ii) hiring j with price p_j will not break the investment constraint. That is, p_j must be equal to p_h if $\delta(j) = \delta(h)$, or $p_j/\delta(j) \geq p_h/\delta(h)$ otherwise.

Afterwards, $M[h][b][p_h] = M[h][b][p_h] + M[j^*][b - p_h][p_{j^*}]$. To find $M[j^*][b - p_h][p_{j^*}]$ and thus compute the final $M[h][b][p_h]$, we iterate each pair of j and p_j (Lines 10 to 18) where Lines 11 and 17 ensure the investment-persuasive, and Lines 13 and 18 ensure that $M[h][b][p_h]$ is correct. After the computation of matrix M, we find the greatest value among all the entries $M[\cdot][B][\cdot]$ and subtract this value from B to obtain an optimal value with respect to Objective (1).

Example 2. *Suppose* $F = \{v_1, v_2\}$ *and* $B = 280$, *where* $\delta(1) = 100$, $p^1_{ask} = 80$, $R_1 = \{80, 100\}$, $\delta(2) = 200$, $p^2_{ask} = 200$ *and* $R_2 = \{140, 200\}$. *Then* $M[1][b][80] = f(1, 80) = 80$ $(80 \leq b \leq B)$ *and* $M[1][b][100] = f(1, 100) = 60$ $(100 \leq b \leq B)$. *To compute* $M[2][B][140]$, *the highest possible* $f()$ *score when we have budget B and hire* v_2 *with* \$140 *as the last influencer, we need to consider hiring* v_1 *while maintaining an investment-persuasive campaign. Thus, the legit hiring price of* v_1 *can be* \$80 *or* \$100. *Therefore,* $M[2][B][140] = f(2, 140) + max(M[1][B - 140][100], M[1][B - 140][80]) = 220$. *For* $M[2][B][200]$, *the only legit price for hiring* v_1 *is* \$100, *which exceeds the remaining budget* 80. *Thus,* $M[2][B][200] = f(2, 200) = 200$. *By searching all entries* $M[\cdot][B][\cdot]$, $M[2][B][140] = 220$ *is the maximum and the optimal divergence is* $280 - 220 = 60$.

Algorithm 4: CR-Inf

Input : A set F of candidates where each candidate v has an asking price p^v_{ask} and the input acceptable price range R_v, and $B = \sum_{v \in F} p^v_{ask}$.
Output : Divergence score.

1 Sort each $v \in F$ based on $\delta(v)$ in non-increasing order.;
2 $S = \emptyset$, $b = B$;
3 Initialize the campaign $\mathcal{P}$ where $\mathcal{P}(v) = 0$, $\forall v \in F$;
4 **foreach** $v \in F$ **do**
5   Update R_v;
6   **if** $Q_v = R_v \cap [0, b] \neq \emptyset$ **then**
7         $p_v = BestPrice(Q_v)$;
8         $\mathcal{P}(v) = p_v$; $b = b - p_v$; $S = S \cup \{v\}$;
9   **if** $b = 0$ **then** Break;
10 **return** $B - \sum_{v \in S} f(v, p_v)$;

Running Time Analysis. Despite its effectiveness, computing the matrix M incurs *pseudo*-polynomial time complexity of $O(|F|^2 \cdot |R_{\max}|^2 \cdot B)$, where $R_{\max}$ is the largest price choice set.

7 PDMIC WITH CONTINUOUS PRICE RANGES

In this section, we further relax the previous restrictions on the price choices such that we set the hiring price of each influencer v as 0 or any price in the acceptable price range $R_v = [\tau(v), B]$. According to the hardness result in Section 4, it is unlikely that efficient exact algorithms for PDMIC could be found. Next, we introduce two heuristic algorithms which do not achieve any non-trivial approximation guarantees but are highly effective in practice, as confirmed by our experimental results.

7.1 An Influence based Greedy Method

Our first heuristic algorithm is called *Continuous-Range and Influence based Method* (CR-Inf), which is a greedy method by prioritizing the satisfaction of influencers with high influence. The pseudo code is shown in Algorithm 4. The intuition behind this algorithm is that influencers with higher influence usually have higher asking prices. Thus, prioritizing the satisfaction of top influencers will potentially result in a large contribution to the total $f()$ score.

Suppose we sort all the influencers by their influence in a *non-increasing* order and break ties by considering smaller asking prices first. The basic idea of the CR-Inf algorithm is to iteratively assign a price to an influencer v based on this order. In each iteration, it updates and maintains R_v for the current candidate v. Specifically, R_v is initialized as $[\tau(v), B]$ and updated based on the current selected influencer set S such that any price in R_v for influencer v to be chosen will not violate the criteria of investment-persuasive campaigns (Line 5). If $Q_v = R_v \cap [0, b] \neq \emptyset$ (i.e., there exists a price for v that can be chosen without violating either the budget or the investment-persuasive constraint), it adds v to S, assigns to v the price p_v in Q_v that is *closest* to v's asking price (i.e., $\min_{p_v \in Q_v} |p_v - p^v_{ask}|$), and

Algorithm 5: WeightedSubsequenceScore (WSS)

Input : A set F of candidates where each candidate v has an asking price p^v_{ask} and the input acceptable price range R_v.

Sort each $v \in F$ based on $\delta(v)$ in non-increasing order;
Initialize M where $M[h] = f(h, p^h_{ask}) = p^h_{ask}$ if p^h_{ask} is in R_h; otherwise, $M[h] = 0$;
for $h = 1$ *to* $|F|$ **do**
  for $j = 1$ *to* $h - 1$ **do**
    if $M[j] + p^h_{ask} > M[h]$ **then**
      if $\delta(j) = \delta(h)$ *and* $p^j_{ask}/\delta(j) = p^h_{ask}/\delta(h)$ **then**
        $M[h] = M[j] + p^h_{ask}$;
      if $\delta(j) > \delta(h)$ *and* $p^j_{ask}/\delta(j) \le p^h_{ask}/\delta(h)$ **then**
        $M[h] = M[j] + p^h_{ask}$;
return M;

Algorithm 6: WeightedSubsequence (WS)

Input : A set F of candidates, subsequence score matrix M, sequence score s, the end index h, the solution set S.

if $s > 0$ **then**
  $id = h$;
  if $id = |F|$ **then**
    while $M[id] \neq s$ **do** $id = id - 1$;
  else
    for $j = h - 1$ *to 1* **do**
      if $M[j] = s$ **then**
        $bool1 \leftarrow \delta(h) = \delta(j)$ and $p^h_{ask} = p^j_{ask}$;
        $bool2 \leftarrow \delta(h) \neq \delta(j)$ and $p^j_{ask}/\delta(j) \le p^h_{ask}/\delta(h)$;
        if $bool1$ *or* $bool2$ **then** $id = h$; Break;
  $p_{id} = p^{id}_{ask}$; Add id into S;
  $WeightedSubsequence(F, M, s - p^{id}_{ask}, id, S)$;

updates $b \leftarrow b - p_v$ (Lines 6 to 8). This iterative process terminates when no more influencers could be selected.

Running Time Analysis. Considering that the influencers are selected with their influence in a non-increasing order. to avoid violating the investment-persuasive constraint, influencers must be selected with price rate non-decreasingly. As a result, when an influencer v is checked, the feasible price range R_v can be easily calculated in $O(1)$ time by the price rate of the most recent selected influencer; and Q_v is just simply $R_v \cap [0, b]$. Therefore, the processing cost for checking each influencer is bounded by $O(1)$ and hence, $O(|F|)$ in total. Plus the sorting cost at the beginning, the overall running time is bounded by $O(|F| \log |F|)$.

Limitations. Despite its high efficiency, CR-Inf tends to satisfy asking prices of top influencers, making it often lack of a global view. And hence, it may fail to consider the impact brought by the investment-persuasive constraint. If the price rates of influencers chosen in S are very high, the price rates of subsequent influencers must be set higher to ensure the campaign is investment-persuasive. If the asking prices of subsequent influencers (in the sorted list) are much lower than the prices that CR-Inf assigns to them, these influencers will be overly satisfied, and the budget is "wasted". The excessive budget spent on these influencers actually brings no increments to the total $f()$ score. This strategy will cause a domino effect such that the budget could be "wasted" significantly and thus many influencers cannot even be hired.

Algorithm 7: CR-MWS

Input : A set F of candidates where each candidate v has an asking price p^v_{ask} and the input acceptable price range R_v, and $B = \sum_{v \in F} p^v_{ask}$.

Output: Divergence score.

1 Sort each $v \in F$ based on $\delta(v)$ and then $\delta(v)/p^v_{ask}$ in non-increasing order, and relabel their IDs from 1 to $|F|$ in this order;
2 $S = \emptyset$;
3 $M = WeightedSubsequenceScore(F)$;
4 $WeightedSubsequence(F, M, \max(M), |F|, S)$ // Obtain an ordered subsequence S of F that achieves the maximum score $\max(M)$ in M;
5 $B_r = B - \max(M)$; **if** $B_r = 0$ **then return** $B - \sum_{v \in S} f(v, p_v)$;
6 $l = \min(S)$, $r = \max(S)$ // Obtain the smallest and largest ID in S;
7 **for** $h = 1$ *to* $l - 1$ **do** $p_h = \delta(h) \cdot p^l_{ask}/\delta(l)$;
8 **for** $h = r + 1$ *to* $|F|$ **do** $p_h = \delta(h) \cdot p^r_{ask}/\delta(r)$;
9 **for** $j = 1$ *to* $|S| - 1$ **do**
10   $l = S(j)$; $r = S(j + 1)$ // Two consecutive influencers in S;
11   **if** $r - l < 2$ **then** break; // No segment between l and r;
12   $ub = p^r_{ask}/\delta(r)$; $lb = p^l_{ask}/\delta(l)$; $preInf = \delta(r)$;
13   **for** $h = (r - 1)$ *to* $(l + 1)$ **do**
14   **if** $lb = ub$ *or* $\delta(h) = \delta(l)$ **then**
15   **if** $lb \cdot \delta(h) \in R_h$ **then** $p_h = lb \cdot \delta(h)$;
16   **else** $p_h = 0$ // h does not have a feasible price range ;
17   Continue;
18   **if** $\delta(h) = preInf$ **then**
19   **if** $ub \cdot \delta(h) \in R_h$ **then** $p_h = ub \cdot \delta(h)$;
20   **else** $p_h = 0$;
21   Continue;
22     $range = [lb * \delta_h, ub * \delta_h] \cap R_h$;
23   **if** $range \neq \emptyset$ **then**
24   **if** $p^h_{ask}/\delta(h) < p^l_{ask}/\delta(l)$ **then** $p_h = min(range)$;
25   **else** $p_h = max(range)$;
26       $ub = p_h/\delta(h)$;
27       $preInf = \delta(h)$;
28   **else** $p_h = 0$;
29 $F_{re} = \{h | h \in F \setminus S$ and $p_h > 0\}$;
30 sort each $v \in F_{re}$ based on $f(v, p^v_{ask})/p_v$ in descending order;
31 **foreach** $v \in F_{re}$ **do**
32   // Select the rest influencers greedily
33   **if** $p_v \le B_r$ **then** $S = S \cup \{v\}$; $B_r = B_r - p_v$;
34 Initialize the campaign $\mathcal{P}$: $\forall v \in F$, $\mathcal{P}(v) = 0$;
35 **foreach** $v \in S$ **do** $\mathcal{P}(v) = p_v$;
36 **return** $B - \sum_{v \in S} f(v, p_v)$;

7.2 A Weighted-Subsequence based Method

To further improve the effectiveness, we propose a *Continuous-Range and Maximum-Weighted-Subsequence based Method* called (CR-MWS). Observe that the price rate deviation of each influencer from her asking one can lead to: this influencer being either *overly satisfied* or *unsatisfied*. In the former case, some budgets would be "wasted". In the latter case, every unit budget spent on the current influencer contributes to the increment of the $f()$ score. Thus, we need to carefully decide the order of satisfying influencers to avoid notable price deviation and budget "abuse".

Algorithm 7 describes its overall procedure. The CR-MWS algorithm consists of three steps, namely *maximum weighted subsequence discovery* (Lines 3 to 4), *price rate adjustment* (Lines 5 to 28) and *candidate finalization* (Lines 29 to 35). In step 1, some influencers are hired by paying for their asking prices; in step 2, the price rates of the remaining influencers are adjusted based on those hired ones; in step 3, among the remaining influencers, who to be hired is determined based on the adjusted price rates.

Step 1: Maximum-weighted subsequence discovery. CR-MWS first sorts influencers by their influence in a *non-increasing* order.

Next, it finds a subsequence of influencers, selecting whom would bring the *maximum* increment of the total $f()$ score without compromising the prices from their asking prices while still meeting the criteria of an investment-persuasive campaign. Such a subsequence is called the *maximum weighted subsequence* (MWS) and its $f()$ score is computed by our proposed method, called *Weighted Subsequence Score* (WSS) (Algorithm 5). Here, influencers in any subsequence are sorted non-increasingly w.r.t. influence.

Specifically, WSS uses a matrix M to record the results where $M[h]$ stores the total $f()$ score of the *local* maximum weighted subsequence among all *qualified* weighted subsequences ended with influencer h. We say a subsequence is *qualified* if hiring all influencers in this subsequence with their asking prices does not violate the investment-persuasive constraint. If p^h_{ask} is in the input price range, the entry $M[h]$ is initialized as p^h_{ask}; otherwise, it is 0. If $M[h] = p^h_{ask}$, it indicates that the local maximum weighted subsequence ended with h currently only contains influencer h. To update $M[h]$, we need to append influencer h to a weighted subsequence which (i) ends with an influencer j ranked before h (i.e., $1 \leq j < h$), (ii) can include h without breaking the investment-persuasive constraint, and (iii) has the greatest $f()$ score among all subsequences satisfying the previous two requirements. Once we find such an entry $M[j]$, we set $M[h] = p^h_{ask} + M[j]$. Lines 4-9 in Algorithm 5 describes the process of updating $M[h]$. After we update the $f()$ scores for all entries in M, we find the greatest entry $\max(M)$ which records the $f()$ score of the *global* MWS.

Afterwards, we propose a backtracking process called *WeightedSubsequence* (i.e., invoked in Line 4 of Algorithm 7), to retrieve all influencers in the maximum weighted subsequence based on $\max(M)$ and store them into the initially empty solution set S. These influencers are hired with their asking prices.

Step 2: Price rate adjustment. In this step, we adjust the price rates of the influencers in $F \setminus S$ to ensure the investment-persuasive constraint if they are hired. Lines 5 to 28 in Algorithm 7 describes the adjustment process and we describe its details below.

Observe that by removing the influencers in the global maximum-weighted subsequence S, the influencers in $F \setminus S$ are divided into "segments" (in the sorted order). Given that all the influencers in S are hired, to ensure an investment-persuasive campaign, we adjust the price rates for the rest influencers in each segment in $F \setminus S$ as follows. Let $\min(S)$ and $\max(S)$ be the smallest and largest influencer ID's in S. Firstly, for each influencer h in the possibly existing segment $[1, \min(S) - 1]$ ($[\max(S) + 1, |F|]$), she will be unsatisfied (overly satisfied) if hired. Thus, we need to set her hiring price as *high* (*low*) as possible, and thus assign her with the same price rate as the influencer $\min(S)$ ($\max(S)$), as shown in Lines 7 and 8.

Secondly, for each segment $[l+1, r-1]$ defined by two consecutive influencers l and r in S, we enforce the price rates of the ordered influencers in this segment to be in non-decreasing order and in the range of $[lb = p^l_{ask}/\delta(l), ub = p^r_{ask}/\delta(r)]$, where lb and ub denote the lower and upper bounds of the hiring price rate respectively. Such non-decreasing price rates can be obtained by deciding the hiring price for each influencer one-by-one. There are two possible ways: (i) the *forward* way: decide the price for $l + 1$ first and all the way to $r - 1$; and (ii) the *backward* way: decide the price for $r - 1$ first and all the way to $l + 1$. Based on our experimental results, the backward way is more effective, and hence, we adopt this direction. Intuitively, the backward way tends to increase the prices for unsatisfied influencers, as shown later in Example 3.

Figure 2: A running example of CR-MWS

More specifically, given an influencer h in a segment, we first compute its qualified price range $[lb \cdot \delta_h, ub \cdot \delta_h] \cap R_h$ (Line 22). If the asking price rate is smaller than $p^l_{ask}/\delta(l)$ (i.e., h will be overly satisfied if hired), we assign h with the minimum price from the qualified price range (Line 24). Otherwise, we assign h with the maximum price (Line 25). Afterwards, update the upper bound ub to be the hiring price rate of h (Line 26). Besides, there are two edge cases where we need to decide the prices more carefully. The first case is when $lb = ub$ or $\delta(h) = \delta(l)$, we need to check if h can be assigned with price $lb \cdot \delta(h)$ (i.e., Lines 14 to 17). The second case is when h has the same influence (i.e., $preInf$ computed in Line 28) as the influencer most recently assigned with a price, we need to check if h can be assigned with the same price (i.e., Lines 18 to 21).

Step 3: Candidate finalization. Once CR-MWS adjusts the prices for hiring the remaining influencers, the third step is to greedily choose them based on the $f()$ score gain per unit adjusted price until the budget is exhausted or all remaining influencers have been processed (Lines 29 to 35 in Algorithm 7).

EXAMPLE 3. *Figure 2 shows the pricing strategy of Algorithm 7 for six influencers. Here, the hiring price threshold $\tau(v) = 0$ for all v, budget $B = 800$, and influencers v_4 to v_6 have the same influence and asking price. In Step 1, we need to find the Maximum Weighted Subsequence (MWS). The subsequence formed by v_1 and v_2 is not a qualified candidate because v_1 has larger influence but a higher asking price rate. On the other hand, the subsequence formed by v_1 and v_3 is qualified. Among the $f()$ scores of all the qualified subsequences, $M[6] = 500$ is the largest. Then, we invoke* WeightedSubsequence *(Line 4 of Algorithm 7) to collect those influencers (in the MWS achieving $M[6]$) into S. In Step 2, after identifying the MWS, we adjust the price rates of the remaining influencers (i.e., v_2 and v_3) in the segment defined by v_1 and v_4 in a backward way. As a result, both v_2 and v_3 have the same adjusted price rate of* 0.4 *and adjusted price of \$200. However, due to the remaining budget limit (of \$300), only v_3 is hired as it has higher $f()$ score per price unit. Thus, the total $f()$ score is $700 = 500 + 200$ contributed by the MWS and v_3 respectively.*

Running Time Analysis. Sorting the influencers takes $O(|F| \log |F|)$ time. The first step runs in $O(|F|^2)$ time, and the second and third step both run in $O(|F|)$ time. The total time complexity is $O(|F|^2)$.

Table 4: Data statistics

Dataset	User Size	Total Degree	Max. Degree	Avg. Degree
LastFM	7.6K	55.6K	216	7.29
Dogster	426.8K	17.1M	46.5K	40.0
Flixster	2.5M	15.8M	1.5K	6.3
Orkut	3.1M	234.4M	33.3K	76.3

8 EXPERIMENTS

In this section, we conduct extensive experiments on real-world datasets to demonstrate the effectiveness and efficiency of our proposed methods. In Section 8.1, we introduce some interesting evaluation metrics. Then we describe the experimental setup in Section 8.2. Afterwards, we present the experimental result in Section 8.3.

8.1 Evaluation Metrics

Since there is no prior work on this topic, we carefully design the evaluation metrics by considering the interests of all stakeholders. We believe these metrics would be useful to future efforts on this topic. Specifically, we use the **Divergence Ratio (D-Ratio)** to evaluate whether a campaign $\mathcal{P}$ can maintain a good balance between conflicting interests of the brand and the influencers on the cost: **D-Ratio** $= \mathcal{D}_{\mathcal{P}}/B$, where $\mathcal{D}_{\mathcal{P}}$ refers to the divergence score of this campaign (see Definition 1) and B is the input budget from the brand. Observe that the budge B actually also represents the worst possible profit divergence. The D-ratio measures how "far" a method's effectiveness is away from the worst solution.

To evaluate the advertising exposure achieved by a campaign, we use the **Influence Ratio (I-Ratio)** to estimate the influence: **I-Ratio** $= \delta_{\mathcal{P}}/OPT_{inf}$, where $\delta_{\mathcal{P}} = \sum_{v \in F \wedge \mathcal{P}(v)>0} \delta(v)$ and OPT_{inf} refers to the optimal influence with the influence maximization objective[1]. In the influence maximization objective, the cost for hiring an influencer is the standard price rate times her influence. Thus, this optimal influence approximates the brand's estimation on the *ideal* advertising exposure effect. For simplicity, we assume the standard price rate to be *1* since it does not impact the methods' effectiveness. Thus, to compute the optimal influence, we just set the $f()$ score of each influencer to be her influence in BC-Exact.

To evaluate whether a campaign is able to bring considerable investment which impacts the agency's commission and the number of potential influencers being involved, we use the **Invested Budget Ratio (B-Ratio)** to measure the invested budget: **B-Ratio** $= B_{\mathcal{P}}/B$, where $B_{\mathcal{P}} = \sum_{v \in F} \mathcal{P}(v) \leq B$. Due to the timeliness of marketing, efficiency is also important. Thus, we report the **Running Time** to build a campaign.

8.2 Experimental Setup

Datasets. We use four real-world social network datasets [45] whose statistics are shown in Table 4. It serves the purpose to use only the degree of users in each dataset.

Categorization of Influencers. To test the robustness of our methods to candidate sets with different degrees of distribution, we coarsely divide influencers into three classes, *Macro*, *Micro* and *Nano*, with the naming convention in this domain [6]. Specifically, we order online users in non-increasing order of their degrees, and then group influencers based on this order into the same tier (i.e., Macro, Micro or Nano influencer) if their total degrees constitute 20% of the total degrees in the social network, as shown in Figure 3.

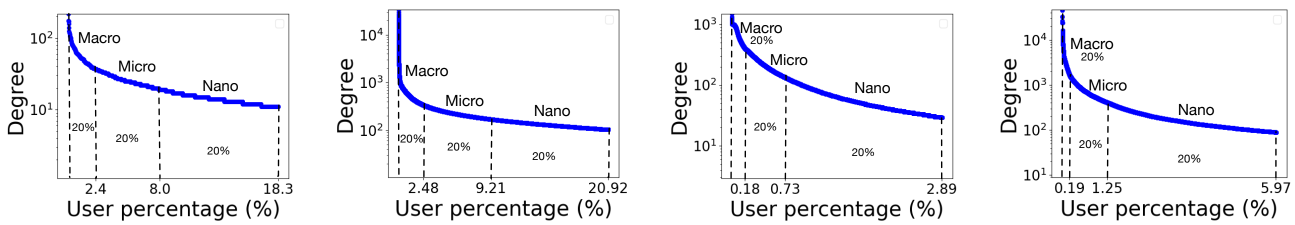
Methods for comparison.

- Binary-Choice based methods, BC-Exact and BC-MG, which can only hire an influencer with the cost equal to her influence or not hire her with no cost. Notably, BC-MG is an extension of a general strategy in resource allocation, as discussed in Section 2.
- Integer-Choice based method IC-Exact which finds the optimal solution when each influencer v is assigned with an *integer* price choice set R_v. Note that it can also be used as a *near-optimal* solution to the PDMIC problem that allows choosing any price in the input acceptable price range if R_v is sufficiently large. Specifically, R_v consists of her influence plus a number, *num*, of integer prices evenly dividing the continuous range $[0.5 * \delta(v), 1.5 * \delta(v)]$ in a coarse-grained level. That is, R_v consists of the integers which are floors of floats in $\{\delta(v)(0.5 + 1/num), \delta(v)(0.5 + 2/num), \ldots, 1.5\delta(v)\}$, where $|R_v|$ is set as 10 by default. If $R_v \not\subseteq [\tau(v), B]$, $R_v = R_v \cap [\tau(v), B]$. Note that this continuous range to be divided is decided based on our many testings which show that a greater range with the same $|R_v|$ will incur much more computational cost without helping IC-Exact produce better solutions.
- Continuous-Range based methods CR-Inf and CR-MWS which can choose any price in the acceptable price range $[\tau(v), B]$ to hire each influencer v. Notably, CR-Inf is an extension of a general strategy in resource allocation, as discussed in Section 2.

Parameter settings.

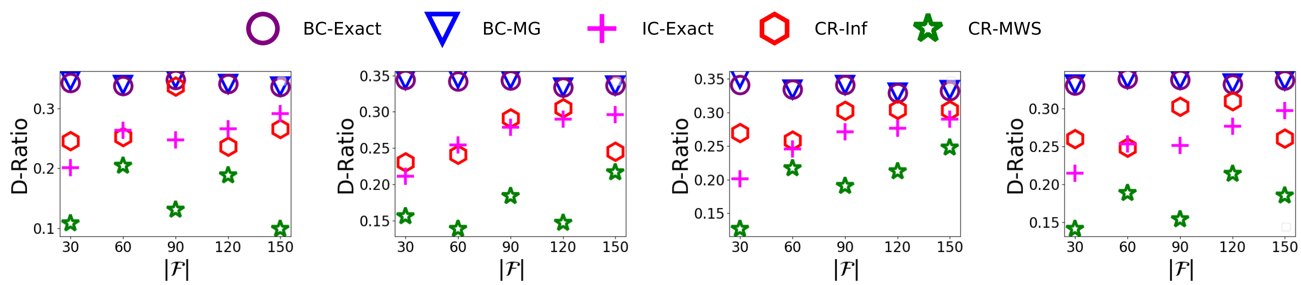
- **Hiring price thresholds**. Considering that the solution space is heavily impacted by the acceptable price range $[\tau(\cdot), B]$ and our objective is to effectively and efficiently solve the PDMIC for any instance, we tackle the largest possible search space by setting the price thresholds of all influencers as 0, which can reflect the efficiency and accuracy gaps among the methods to the maximum extent. Furthermore, with this setup, all the inputs naturally satisfy $\delta(v) \geq \tau(v) = 0$ for any candidate influencer v and hence the restriction (i.e., $\delta(v) \geq \tau(v)$) of BC-Exact and BC-MG no longer exists.
- **The investment budget**. The brand offers a budget to the agency. The agency will find suitable candidate influencers who have high alignments with the brand's product to be promoted, and whose total influence should match with the budget considering the standard price rate (i.e., 1 in experiments) in this market [16]. The final budget B is set by a negotiation between the brand and the agency based on candidates being found.
- **The candidate influencers**. Here, we assume the candidate set has been found since this process is orthogonal to our study. Considering the charges on contracted influencers for agent commission (usually 20% [26]) and that it may not be possible to find suitable candidates perfectly matched with the budget, we create the candidate set F by randomly selecting a number of candidates ($|F| = 90$ by default) from the aforementioned three tiers of influencers (i.e., *Macro*, *Micro* and *Nano*), and set B as a percentage β (80% by default) of the total influence of candidates times the standard price rate (i.e., 1). To show the robustness of our methods to the combinations of influencers of different categories, we create

[1] We assume there is barely influence overlap between influencers found in Step 2 in Figure 1, as it can be easily detected, preprocessed and orthogonal to our study.

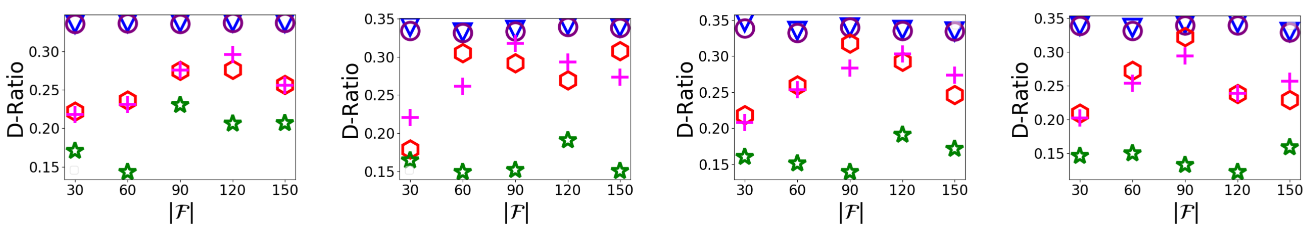


(a) LastFm (b) Orkut (c) Flixster (d) Dogster

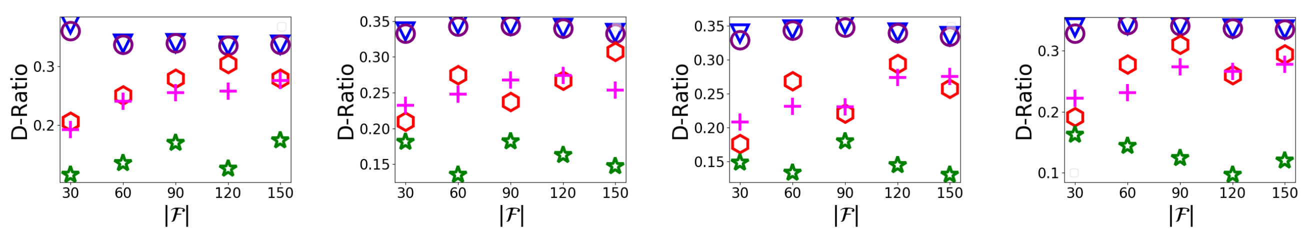
Figure 3: Degree distributions and influencer categorization.



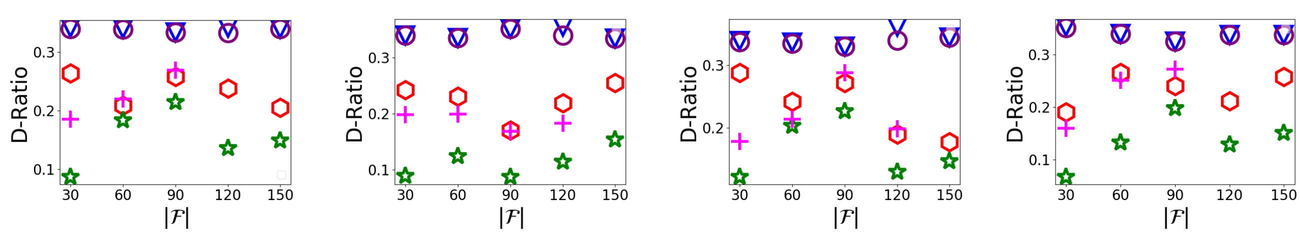
(a) Macro-focused sampling on LastFM (b) Micro-focused sampling on LastFM (c) Nano-focused sampling on LastFM (d) Uniform sampling on LastFM



(e) Macro-focused sampling on Orkut (f) Micro-focused sampling on Orkut (g) Nano-focused sampling on Orkut (h) Uniform sampling on Orkut



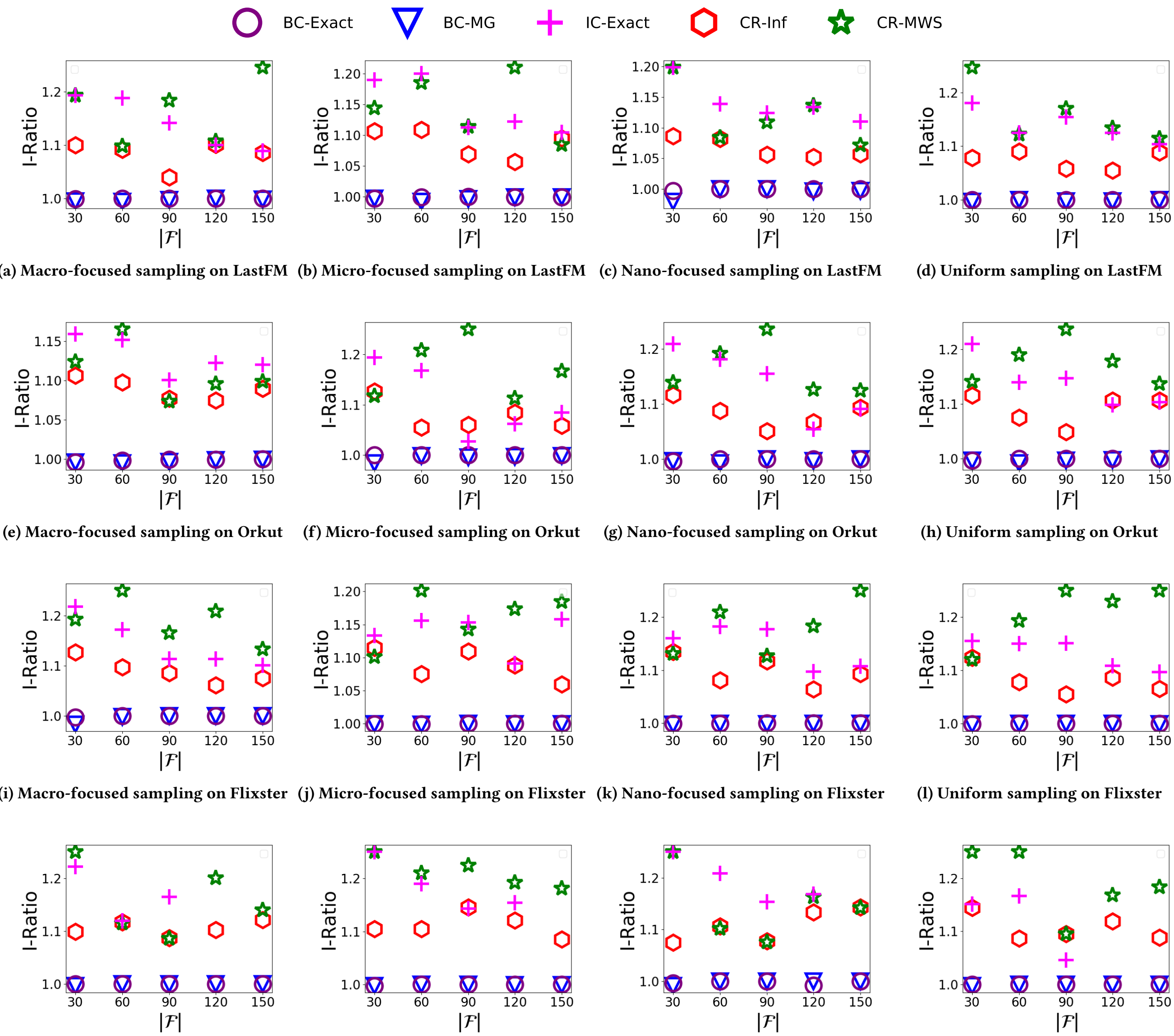
(i) Macro-focused sampling on Flixster (j) Micro-focused sampling on Flixster (k) Nano-focused sampling on Flixster (l) Uniform sampling on Flixster



(m) Macro-focused sampling on Dogster (n) Micro-focused sampling on Dogster (o) Nano-focused sampling on Dogster (p) Uniform sampling on Dogster

Figure 4: The divergence ratio comparison. Methods are more effective with lower ratios. (Note: there is no trend to show for each method when $|F|$ increases since cases are independent. Hence, line chart is not used.)

the candidate set with the following different sampling distributions: {Macro:0.5, Micro:0.25, Nano:0.25}, {Macro:0.25, Micro:0.5, Nano:0.25}, {Macro:0.25, Micro:0.25, Nano:0.5}, and {Macro:0.33,



(a) Macro-focused sampling on LastFM (b) Micro-focused sampling on LastFM (c) Nano-focused sampling on LastFM (d) Uniform sampling on LastFM

(e) Macro-focused sampling on Orkut (f) Micro-focused sampling on Orkut (g) Nano-focused sampling on Orkut (h) Uniform sampling on Orkut

(i) Macro-focused sampling on Flixster (j) Micro-focused sampling on Flixster (k) Nano-focused sampling on Flixster (l) Uniform sampling on Flixster

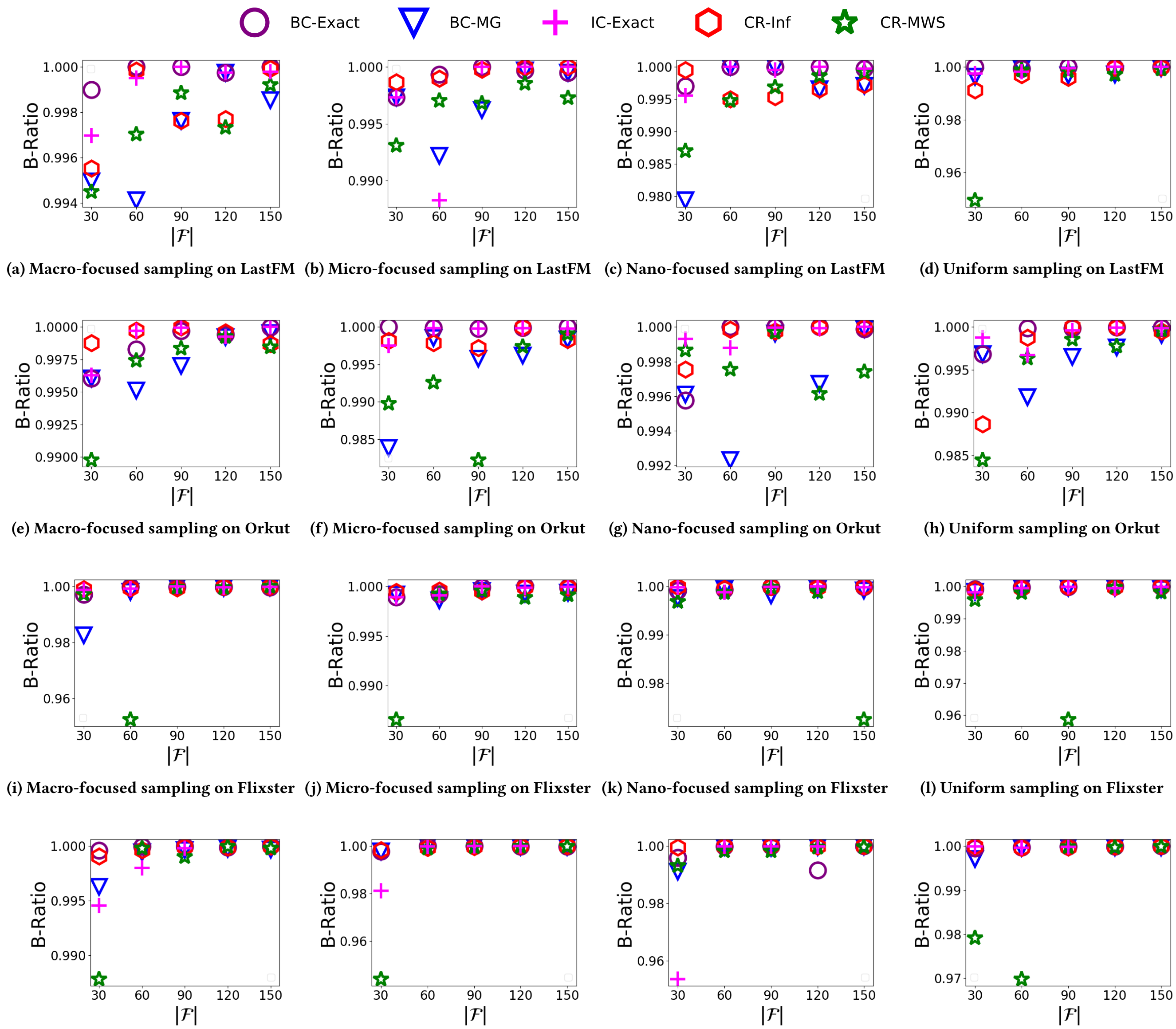
(m) Macro-focused sampling on Dogster (n) Micro-focused sampling on Dogster (o) Nano-focused sampling on Dogster (p) Uniform sampling on Dogster

Figure 5: The influence ratio comparison. Methods are more effective with higher ratios. (Note: there is no trend to show for each method when $|F|$ increases since cases are independent. Hence, line chart is not used.)

Micro:0.33, Nano:0.33}. For simplicity, we call them as Macro-focused, Micro-focused, Nano-focused and Uniform sampling distributions, respectively. Due to the page limit and consistent observations in different distributions, in the experiment, we create the candidate set F with only the Uniform and Micro-focused sampling distributions.

- **Asking price**. Since each influencer v values her own influence differently in reality, depending on concrete campaign contract and personal factors (e.g., exclusive collaboration with this brand, likeness towards the product to be promoted, choices of media channels, etc.) [17, 18], we use the weighted influence sampled from a consecutive integer range to include these realistic factors. Considering that influencers usually overprice their influence and following the findings reported in [25], we use the range $[\delta(v), 1.5\delta(v)]$ for sampling. The asking price controlled by the agency is a weighted-influence related *percentage* over the budget, and we compute this *percentage* as the weighted influence of the influencer over the total one of all candidates.

Environments. We conduct all experiments on a Linux server with Intel Xeon E5 (2.60 GHz) CPUs and 512 GB RAM. All codes are implemented in Python and any method which cannot finish within 60 hours will be terminated. Our code is available at [24].



(a) Macro-focused sampling on LastFM (b) Micro-focused sampling on LastFM (c) Nano-focused sampling on LastFM (d) Uniform sampling on LastFM

(e) Macro-focused sampling on Orkut (f) Micro-focused sampling on Orkut (g) Nano-focused sampling on Orkut (h) Uniform sampling on Orkut

(i) Macro-focused sampling on Flixster (j) Micro-focused sampling on Flixster (k) Nano-focused sampling on Flixster (l) Uniform sampling on Flixster

(m) Macro-focused sampling on Dogster (n) Micro-focused sampling on Dogster (o) Nano-focused sampling on Dogster (p) Uniform sampling on Dogster

Figure 6: The invested budget ratio comparison. Methods are more effective with higher ratios. (Note: there is no trend to show for each method when $|F|$ increases since cases are independent. Hence, line chart is not used.)

8.3 Experimental Results

Upon the aforementioned metrics, we first evaluate to which extent different methods protect benefits and improve satisfaction of different stakeholders. Furthermore, we conduct an ablation study on the impact of the size of price choices on IC-Exact to demonstrate the effectiveness of IC-Exact and CR-MWS.

8.3.1 Divergence ratio comparison. Figure 4 compares the divergence ratio w.r.t. different sampling distributions and candidate sizes. We have the following observations: (1) BC-MG shows very competitive performance compared with BC-Exact, thereby demonstrating the empirical effectiveness of BC-MG is way better than what its worst-case approximation ratio indicates. However, both of them are notably outperformed by flexible-choices based methods; (2) with CR-Inf as the reference, the performance of IC-Exact is generally better under the uniform sampling distribution than in others. We suspect that IC-Exact is able to produce better campaigns from the limited price choices when the degree distribution of candidates is sparser. When candidates have similar degrees, IC-Exact requires more fine-grained and similar price choices to achieve lower divergence. As for other methods, we do not see any clear impact of sampling distributions on the performance; (3) CR-MWS significantly outperforms all other methods in all cases, which demonstrates its effectiveness of protecting stakeholders' benefits.

8.3.2 *Influence ratio comparison.* Figure 5 compares the influence ratios of each competitor method over the optimal influence based on the standard price rate. The observations are summarized as following: (1) the influence ratio is above 1.0 in many cases. The reason is that, in the problem setting with the influence maximization objective, the hiring price of an influencer is equal to the influencer's influence and can be higher than the hiring price computed by our algorithms which are flexible on the hiring prices. As a result, our algorithms may be able to hire more influencers and thus achieve larger influence coverages than OPT_{inf}. (2) BC-Exact and BC-MG with our objective achieve very competitive influence compared with BC-Exact with the influence maximization objective. Furthermore, the performance ranking of BC-Exact and BC-MG under this metric can be indicated from their profit divergence scores. For example, when BC-Exact achieves a lower divergence score than BC-MG in Figure 4 (c) with $|F| = 30$, it also achieves greater influence under the same setting. These observations indicate the positive correlations between profit divergence minimization and influence maximization under the binary-choice setting. (3) Integer-Choice and Continuous-Range based methods are able to achieve better influence ratios since they have greater space for price adjustment. While maintaining very competitive divergence scores, IC-Exact tends to hire more candidates than CR-Inf which emphasizes satisfying the influencers with great influence and thus quickly exhausts the hiring budget. IC-Exact is very effective since it makes the best strategy with limited integer price choices. (4) Among all the methods, CR-MWS achieves the greatest influence ratios in most cases while maintaining the lowest divergence scores. It indicates that protecting the benefits of influencers does not necessarily compromise the brand's interests.

8.3.3 *Invested budget ratio comparison.* Figure 6 compares the invested budget ratios of different methods. As mentioned earlier, the invested budget ratio is the least important effectiveness evaluation metric to the agency since protecting benefits of the brand and influencers is the first priority. Even though methods such as BC-Exact and CR-Inf slightly outperform CR-MWS in many cases under this metric, they attract the investment in a way that sacrifices the benefits of the brand or influencers as shown in Figure 4 and Figure 5. Thus, the agency should make comparison between different methods under this metric only when these methods are very competitive in previous evaluation.

8.3.4 *Efficiency comparison.* Figure 7 shows the running time of different methods. BC-MG is the fastest one due to its simple strategy and is up to two-orders-of-magnitude faster than BC-Exact. CR-Inf and CR-MWS are ranked as second and third. Despite the notable difference between their time complexity (i.e., $O(|F|\log(|F|))$ and $O(|F|\log(|F|) + |F|^2)$), CR-MWS is very competitive with CR-Inf practically since candidate sets are usually small. On the other hand, IC-Exact can be up to six-orders-of-magnitude slower than other methods, which makes it infeasible in the real world despite its competitive performance against many methods.

8.3.5 *Comparison with near-optimal solutions.* In order to see the performance limit of IC-Exact and further demonstrate the effectiveness of CR-MWS, we gradually increase the size of R for each influencer in IC-Exact. Figure 8 shows the performance of IC-Exact with different sizes of R, where CR-InfInt refers to the version of CR-Inf allowing integer price choices only. As $|R|$ increases, the performance of IC-Exact notably increases and converges to a stable state on LaftFM but fails to converge on Orkut within the our time limit (i.e., 60 hours). An interesting observation is that, even though IC-Exact significantly outperforms all methods, it is still can be outperformed by CR-MWS. We suspect that it is mainly caused by the fact that IC-Exact can only work with integer price choices, as evidenced by the notable performance difference between CR-Inf and CR-InfInt which may even be outperformed by binary-choice based methods. As we can see from Table 5 and Table 6, the running time of IC-Exact increases drastically as $|R|$ and can be up to eight-orders-of-magnitude slower than other flexible-choice based methods (e.g., comparing with $|F| = 90$ in Figure 7(f) with $|R| = 50$), which makes it infeasible to cater for real-world scenarios.

Table 5: Running time (s) of IC-Exact on LastFM.

Dataset	$\|R\|$				
	10	20	30	40	50
Uniform	1.4E+3	3.7E+3	5.3E+3	5.9E+3	6.5E+3
Micro-focused	1.4E+3	3.9E+3	5.3E+3	6.0E+3	6.2E+3

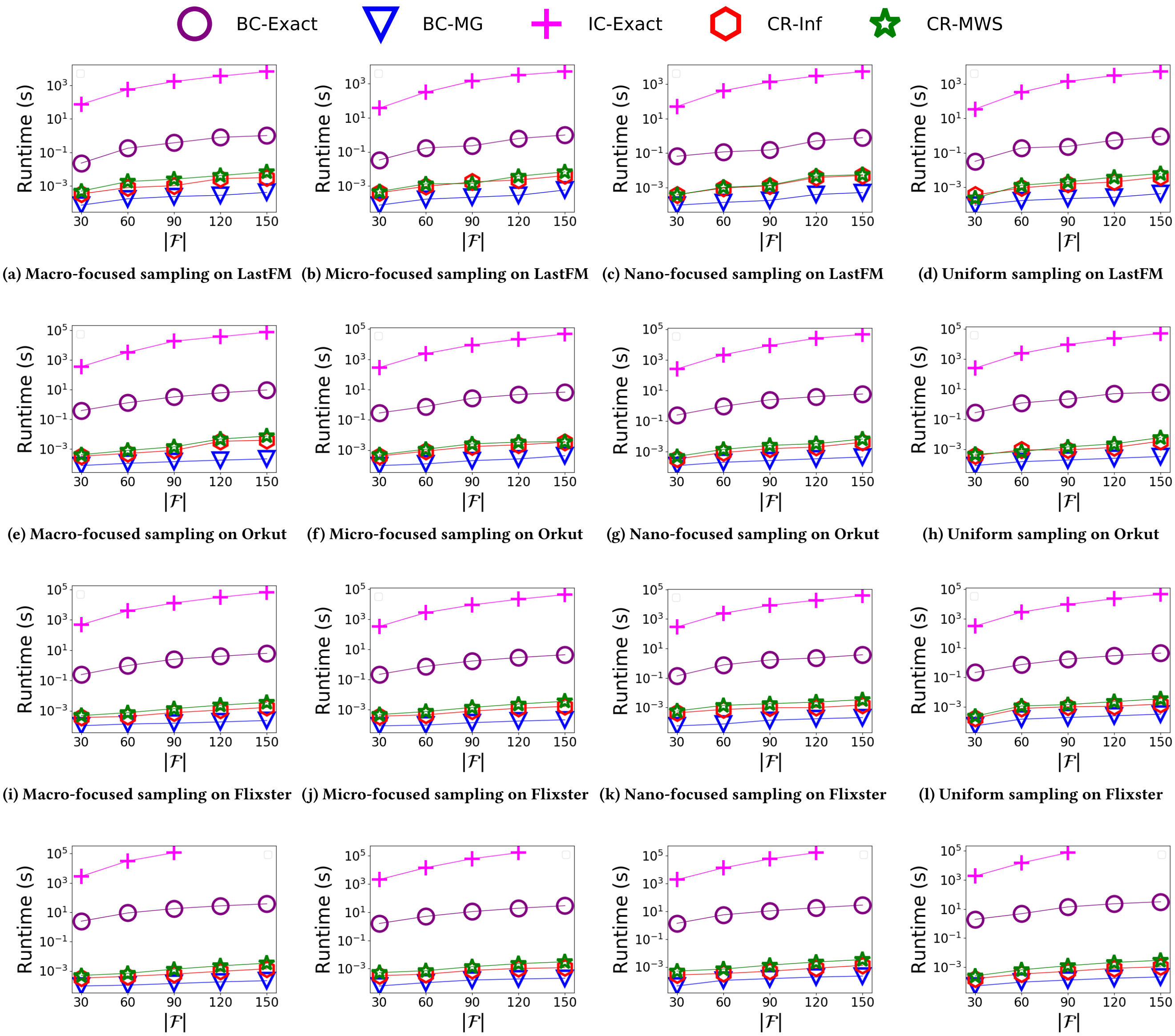
Table 6: Running time (s) of IC-Exact on Orkut.

Dataset	$\|R\|$				
	10	20	30	40	50
Uniform	9.1E+3	3.3E+4	6.9E+4	1.2E+5	1.9E+5
Micro-focused	8.9E+3	3.1E+4	6.7E+4	1.3E+5	1.9E+5

8.3.6 *Ablation study on β.* Recall that the budget B is a percentage β of the total influence of candidates. We study the impact of β on the D-Ratio, and the results are presented in Figure 9. Since BC-Exact, BC-MG and IC-Exact require trial-and-error to set fixed price choices to (efficiently) produce high-quality solutions, their performance based on the default price choice setting for $\beta = 0.8$ may not be effective for a different β and can degrade as β decreases. Since the asking price of an influencer, a *percentage* over the budget, tends to decrease as β decreases and so does the optimal hiring price, the difference between the opimal hiring price and the best price we can choose from the predefined price choices tends to be larger as β decreases, and so does the D-Ratio. On the other hand, CR-Inf and CR-MWS always produce notably better solutions since they consider any price under the budget. Similar to the results in Figure 4, test cases under different settings of β are independent such that the D-Ratio achieved by a method under a specific β cannot be indicated by the ones under other settings of β, and so does the optimal D-ratio. Thus, there is no expected trend of the performance of CR-Inf and CR-MWS. In terms of efficiency comparison, we find that the impact of β is significantly smaller than that of $|F|$ and is barely noticeable in figures, as shown in Figure 10.

9 CONCLUSION

In this paper, we study how minimizing the profit divergence minimization helps build an investment-persuasive influencer marketing campaign, so as to attract investments from the brand while benefiting all stakeholders. We prove this problem to be NP-hard and then propose several methods whose efficiency and effectiveness are demonstrated by our extensive experiments. In future, we



(a) Macro-focused sampling on LastFM (b) Micro-focused sampling on LastFM (c) Nano-focused sampling on LastFM (d) Uniform sampling on LastFM

(e) Macro-focused sampling on Orkut (f) Micro-focused sampling on Orkut (g) Nano-focused sampling on Orkut (h) Uniform sampling on Orkut

(i) Macro-focused sampling on Flixster (j) Micro-focused sampling on Flixster (k) Nano-focused sampling on Flixster (l) Uniform sampling on Flixster

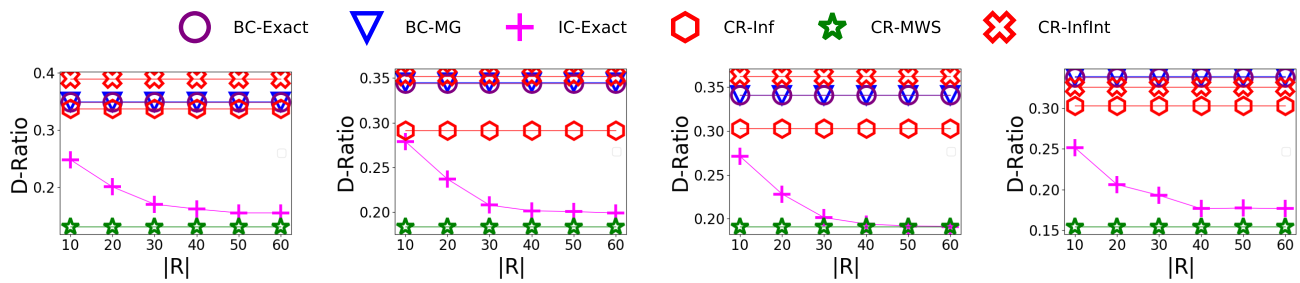
(m) Macro-focused sampling on Dogster (n) Micro-focused sampling on Dogster (o) Nano-focused sampling on Dogster (p) Uniform sampling on Dogster

Figure 7: Running time comparison.

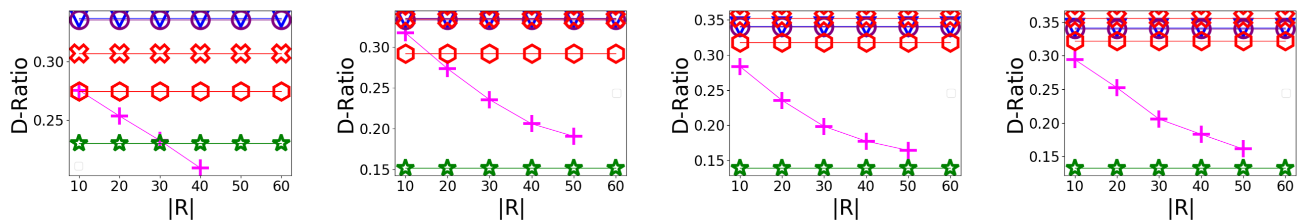
plan to analyze the accuracy of our Continuous-Range based approximate methods theoretically and extend our solutions to other marketing scenarios with similar needs.

REFERENCES

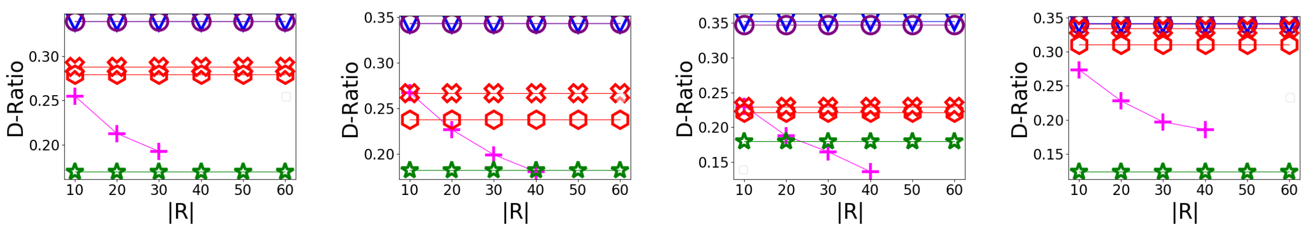
[1] 2009. https://courses.engr.illinois.edu/cs598csc/sp2009/lectures/lecture_4.pdf.
[2] 2016. https://resource.co/article/material-gains-through-brokerage-services-11506.
[3] 2017. https://www.emarketer.com/Article/Influencer-Marketing-Prices-Rising-UK/1016283.
[4] 2019. https://buffer.com/resources/micro-influencers/.
[5] 2019. https://later.com/blog/instagram-influencers-costs/.
[6] 2019. https://bondai.co/blog/6-types-of-influencers/.
[7] 2020. https://socialpubli.com/blog/how-influencer-agency-work/.
[8] 2020. https://influencermarketinghub.com/micro-influencers-vs-celebrities/.
[9] 2020. https://www.oracle.com/a/ocom/docs/cloud/oracle-451-research-advisory-blog-adopting.pdf.
[10] 2020. https://reusewood.org/guide/material-brokerage.
[11] 2020. https://blog.planoly.com/calculate-influencer-rates.
[12] 2021. https://www.marketingdive.com/news/social-commerce-drive-influencer-marketing-evolution-2022/611546/.
[13] 2021. https://www.forbes.com/sites/forbesagencycouncil/2021/06/02/micro-influencers-when-smaller-is-better/?sh=591ee7d1539b.



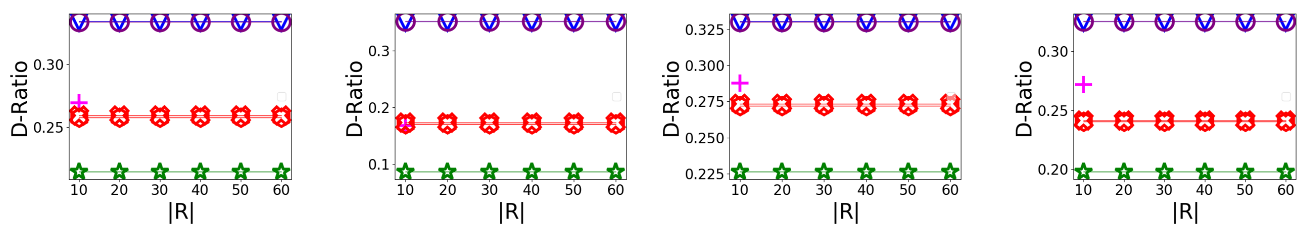
(a) Macro-focused sampling on LastFM (b) Micro-focused sampling on LastFM (c) Nano-focused sampling on LastFM (d) Uniform sampling on LastFM



(e) Macro-focused sampling on Orkut (f) Micro-focused sampling on Orkut (g) Nano-focused sampling on Orkut (h) Uniform sampling on Orkut

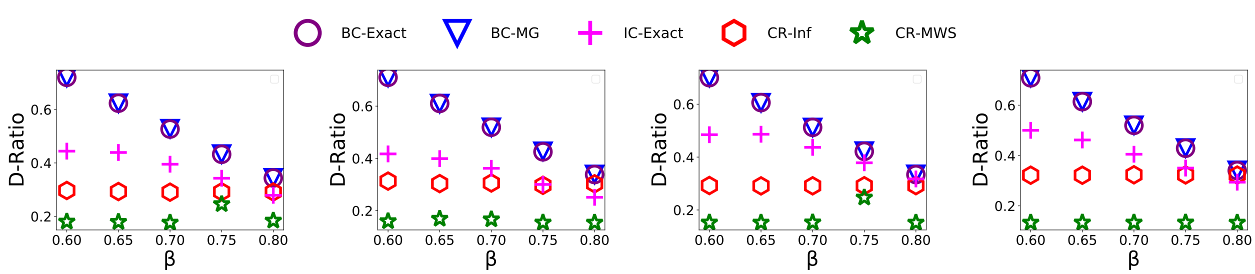


(i) Macro-focused sampling on Flixster (j) Micro-focused sampling on Flixster (k) Nano-focused sampling on Flixster (l) Uniform sampling on Flixster



(m) Macro-focused sampling on Dogster (n) Micro-focused sampling on Dogster (o) Nano-focused sampling on Dogster (p) Uniform sampling on Dogster

Figure 8: Ablation study on $|R|$ in IC-Exact. The price choices of other methods are set by default.

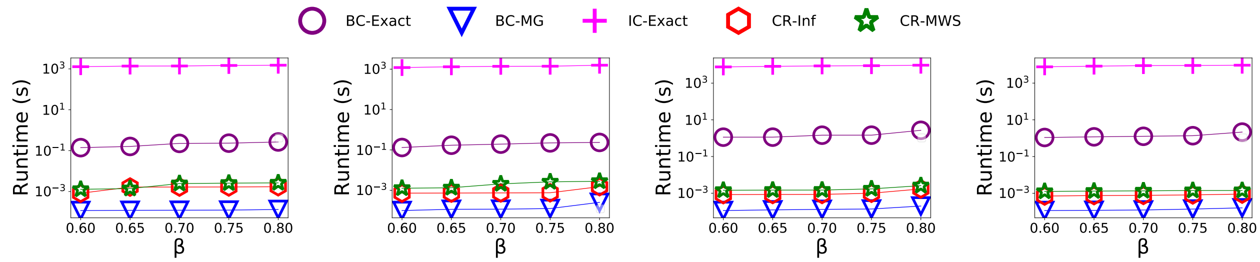


(a) Micro-focused on LastFM (b) Uniform on LastFM (c) Micro-focused on Orkut (d) Uniform on Orkut

Figure 9: Ablation study: The divergence ratio comparison with different β

[14] 2021. https://www.marketingcharts.com/digital/social-media-116631.

[15] 2021. https://www.indeed.com/career-advice/career-development/tiered-pricing-template.



(a) Micro-focused sampling on LastFM **(b) Uniform sampling on LastFM** **(c) Micro-focused sampling on Orkut** **(d) Uniform sampling on Orkut**

Figure 10: Running time comparison with different β.

[16] 2021. https://www.businessofapps.com/marketplace/influencer-marketing/research/influencer-marketing-costs/.
[17] 2021. https://blog.hootsuite.com/instagram-influencer-rates//.
[18] 2021. https://hashtagpaid.com/banknotes/creator-rates.
[19] 2022. https://fourthfloorcreative.co/blog/influencer-marketing-agency.
[20] 2022. https://winkmodels.com.au/much-cost-book-influencer/.
[21] 2022. https://www.britannica.com/topic/operations-research/Resource-allocation.
[22] 2022. https://quickbooks.intuit.com/r/midsize-business/tiered-pricing/.
[23] 2022. https://blog.axway.com/learning-center/cloud-adoption/multi-cloud-strategy.
[24] 2022. https://gitfront.io/r/user-9366410/jA9cWP7WK7Dg/master/.
[25] 2022. https://www.andrewmacarthy.com/andrew-macarthy-social-media/instagram-influencer-pricing.
[26] 2022. https://www.webfx.com/influencer-marketing-pricing.html.
[27] Akhil Arora, Sainyam Galhotra, and Sayan Ranu. 2017. Debunking the myths of influence maximization: An in-depth benchmarking study. In *SIGMOD*. 651–666.
[28] David Arthur, Rajeev Motwani, Aneesh Sharma, and Ying Xu. 2009. Pricing strategies for viral marketing on social networks. In *International workshop on internet and network economics*. 101–112.
[29] Çigdem Aslay, Francesco Bonchi, Laks V. S. Lakshmanan, and Wei Lu. 2017. Revenue Maximization in Incentivized Social Advertising. *Proc. VLDB Endow.* 10, 11 (2017), 1238–1249.
[30] Çigdem Aslay, Wei Lu, Francesco Bonchi, Amit Goyal, and Laks V. S. Lakshmanan. 2015. Viral Marketing Meets Social Advertising: Ad Allocation with Minimum Regret. *Proc. VLDB Endow.* 8, 7 (2015), 822–833.
[31] Suman Banerjee, Mamata Jenamani, and Dilip Kumar Pratihar. 2020. A survey on influence maximization in a social network. *Knowledge and Information Systems* 62, 9 (2020), 3417–3455.
[32] Song Bian, Qintian Guo, Sibo Wang, and Jeffrey Xu Yu. 2020. Efficient algorithms for budgeted influence maximization on massive social networks. *Proceedings of the VLDB Endowment* 13, 9 (2020), 1498–1510.
[33] Christian Borgs, Michael Brautbar, Jennifer Chayes, and Brendan Lucier. 2014. Maximizing social influence in nearly optimal time. In *SODA*. 946–957.
[34] Kurt M Bretthauer and Bala Shetty. 2002. The nonlinear knapsack problem–algorithms and applications. *European Journal of Operational Research* 138, 3 (2002), 459–472.
[35] Taotao Cai, Jianxin Li, Ajmal S Mian, Timos Sellis, Jeffrey Xu Yu, et al. 2020. Target-aware holistic influence maximization in spatial social networks. *TKDE* (2020).
[36] Raul Castro Fernandez. 2022. Protecting Data Markets from Strategic Buyers. In *SIGMOD*. 1755–1769.
[37] Chandra Chekuri and Sanjeev Khanna. 2005. A polynomial time approximation scheme for the multiple knapsack problem. *SIAM J. Comput.* 35, 3 (2005), 713–728.
[38] Lingjiao Chen, Paraschos Koutris, and Arun Kumar. 2019. Towards model-based pricing for machine learning in a data marketplace. In *SIGMOD*. 1535–1552.
[39] Wei Chen, Chi Wang, and Yajun Wang. 2010. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In *SIGKDD*. 1029–1038.
[40] Wei Chen, Yajun Wang, and Siyu Yang. 2009. Efficient influence maximization in social networks. In *SIGKDD*. 199–208.
[41] Wei Chen, Yifei Yuan, and Li Zhang. 2010. Scalable influence maximization in social networks under the linear threshold model. In *ICDM*. 88–97.
[42] Ying-Ju Chen, Yves Zenou, and Junjie Zhou. 2018. Competitive pricing strategies in social networks. *The RAND Journal of Economics* 49, 3 (2018), 672–705.
[43] Suqi Cheng, Huawei Shen, Junming Huang, Wei Chen, and Xueqi Cheng. 2014. IMRank: influence maximization via finding self-consistent ranking. In *SIGIR*. 475–484.
[44] Suqi Cheng, Huawei Shen, Junming Huang, Guoqing Zhang, and Xueqi Cheng. 2013. Staticgreedy: solving the scalability-accuracy dilemma in influence maximization. In *CIKM*. 509–518.
[45] The Koblenz Network Collection. 2017. http://konect.uni-koblenz.de.
[46] Shaleen Deep and Paraschos Koutris. 2017. QIRANA: A framework for scalable query pricing. In *SIGMOD*. 699–713.
[47] Radwa El-Awadi and Mohamed Abu-Rizka. 2015. A framework for negotiating service level agreement of cloud-based services. *Procedia Computer Science* 65 (2015), 940–949.
[48] Sainyam Galhotra, Akhil Arora, and Shourya Roy. 2016. Holistic influence maximization: Combining scalability and efficiency with opinion-aware models. In *SIGMOD*. 1077–1088.
[49] Michael R Garey and David S Johnson. 1979. *Computers and intractability*. Vol. 174. freeman San Francisco.
[50] Jacob Goldenberg, Barak Libai, and Eitan Muller. 2001. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing letters* 12, 3 (2001), 211–223.
[51] Amit Goyal, Wei Lu, and Laks VS Lakshmanan. 2011. Celf++: optimizing the greedy algorithm for influence maximization in social networks. In *WWW*. 47–48.
[52] Amit Goyal, Wei Lu, and Laks VS Lakshmanan. 2011. Simpath: An efficient algorithm for influence maximization under the linear threshold model. In *ICDM*. 211–220.
[53] Mark Granovetter. 1978. Threshold models of collective behavior. *American journal of sociology* 83, 6 (1978), 1420–1443.
[54] Jianxiong Guo and Weili Wu. 2020. Influence maximization: Seeding based on community structure. *TKDD* 14, 6 (2020), 1–22.
[55] Kai Han, Benwei Wu, Jing Tang, Shuang Cui, Cigdem Aslay, and Laks VS Lakshmanan. 2021. Efficient and Effective Algorithms for Revenue Maximization in Social Advertising. In *SIGMOD*. 671–684.
[56] Shuo Han, Fuzhen Zhuang, Qing He, and Zhongzhi Shi. 2014. Balanced seed selection for budgeted influence maximization in social networks. In *PAKDD*. 65–77.
[57] Keke Huang, Jing Tang, Xiaokui Xiao, Aixin Sun, and Andrew Lim. 2020. Efficient approximation algorithms for adaptive target profit maximization. In *ICDE*. 649–660.
[58] Shixun Huang. 2021. *Capturing and leveraging collective behavior for large-scale social networks analysis*. Ph.D. Dissertation. RMIT University.
[59] Shixun Huang, Zhifeng Bao, J.S.Culpepper, and Bang.Zhang. 2019. Finding Temporal Influential Users over Evolving Social Networks. In *ICDE*.
[60] Shixun Huang, Wenqing Lin, Zhifeng Bao, and Jiachen Sun. 2022. Influence maximization in real-world closed social networks. *arXiv preprint arXiv:2209.10286* (2022).
[61] Kyomin Jung, Wooram Heo, and Wei Chen. 2012. Irie: Scalable and robust influence maximization in social networks. In *ICDM*. 918–923.
[62] Mohammad Mehdi Keikha, Maseud Rahgozar, Masoud Asadpour, and Mohammad Faghih Abdollahi. 2020. Influence maximization across heterogeneous interconnected networks based on deep learning. *Expert Systems with Applications* 140 (2020), 112905.
[63] David Kempe, Jon Kleinberg, and Éva Tardos. 2003. Maximizing the spread of influence through a social network. In *SIGKDD*. 137–146.
[64] Sami Khuri, Thomas Bäck, and Jörg Heitkötter. 1994. The zero/one multiple knapsack problem and genetic algorithms. In *Proceedings of the 1994 ACM symposium on Applied computing*. 188–193.
[65] Anton J Kleywegt and Jason D Papastavrou. 1998. The dynamic and stochastic knapsack problem. *Operations research* 46, 1 (1998), 17–35.
[66] Paraschos Koutris, Prasang Upadhyaya, Magdalena Balazinska, Bill Howe, and Dan Suciu. 2013. Toward practical query pricing with querymarket. In *SIGMOD*. 613–624.
[67] Siyu Lei, Silviu Maniu, Luyi Mo, Reynold Cheng, and Pierre Senellart. 2015. Online influence maximization. In *SIGKDD*. 645–654.
[68] Jure Leskovec, Andreas Krause, Carlos Guestrin, Christos Faloutsos, Jeanne VanBriesen, and Natalie Glance. 2007. Cost-effective outbreak detection in networks.

In *SIGKDD*. 420–429.

[69] Jianxin Li, Taotao Cai, Ke Deng, Xinjue Wang, Timos Sellis, and Feng Xia. 2020. Community-diversified influence maximization in social networks. *Information Systems* 92 (2020), 101522.

[70] Lingfei Li, Yezheng Liu, Qing Zhou, Wei Yang, and Jiahang Yuan. 2020. Targeted influence maximization under a multifactor-based information propagation model. *Information Sciences* 519 (2020), 124–140.

[71] Ping Liu, Meng Wang, Jiantao Cui, and Hui Li. 2021. Top-k competitive location selection over moving objects. *Data Science and Engineering* 6, 4 (2021), 392–401.

[72] Ziyang Liu and Hakan Hacigümüs. 2014. Online optimization and fair costing for dynamic data sharing in a cloud data market. In *SIGMOD*. 1359–1370.

[73] Silvano Martello and Paolo Toth. 1980. Solution of the zero-one multiple knapsack problem. *European Journal of Operational Research* 4, 4 (1980), 276–283.

[74] George B Mathews. 1896. On the partition of numbers. *Proceedings of the London Mathematical Society* 1, 1 (1896), 486–490.

[75] Shubhadip Mitra, Priya Saraf, Richa Sharma, Arnab Bhattacharya, and Sayan Ranu. 2019. Netclus: A scalable framework to mine top-k locations for placement of trajectory-aware services. In *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*. 27–35.

[76] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. 1978. An analysis of approximations for maximizing submodular set functions. *Mathematical programming* 14, 1 (1978), 265–294.

[77] Huy Nguyen and Rong Zheng. 2013. On budgeted influence maximization in social networks. *IEEE J-SAC* 31, 6 (2013), 1084–1094.

[78] Naoto Ohsaka, Takuya Akiba, Yuichi Yoshida, and Ken-ichi Kawarabayashi. 2014. Fast and Accurate Influence Maximization on Large Networks with Pruned Monte-Carlo Simulations. In *AAAI*. 138–144.

[79] George Panagopoulos, Fragkiskos D Malliaros, and Michalis Vazirgianis. 2020. Influence maximization using influence and susceptibility embeddings. In *AAAI*, Vol. 14. 511–521.

[80] Pierre Perrault, Jennifer Healey, Zheng Wen, and Michal Valko. 2020. Budgeted online influence maximization. In *ICML*. 7620–7631.

[81] Ulrich Pferschy and Joachim Schauer. 2009. The Knapsack Problem with Conflict Graphs. *J. Graph Algorithms Appl.* 13, 2 (2009), 233–249.

[82] David Pisinger. 1995. A minimal algorithm for the multiple-choice knapsack problem. *European Journal of Operational Research* 83, 2 (1995), 394–410.

[83] Sijie Ruan, Jie Bao, Yuxuan Liang, Ruiyuan Li, Tianfu He, Chuishi Meng, Yanhua Li, Yingcai Wu, and Yu Zheng. 2020. Dynamic public resource allocation based on human mobility prediction. *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies* 4, 1 (2020), 1–22.

[84] Xiaobin Rui, Xiaodong Yang, Jianping Fan, and Zhixiao Wang. 2020. A neighbour scale fixed approach for influence maximization in social networks. *Computing* 102, 2 (2020), 427–449.

[85] Grant Schoenebeck and Biaoshuai Tao. 2020. Influence Maximization on Undirected Graphs: Toward Closing the (1-1/e) Gap. *ACM Transactions on Economics and Computation (TEAC)* 8, 4 (2020), 1–36.

[86] Prabhakant Sinha and Andris A Zoltners. 1979. The multiple-choice knapsack problem. *Operations Research* 27, 3 (1979), 503–515.

[87] Haiqi Sun, Reynold Cheng, Xiaokui Xiao, Jing Yan, Yudian Zheng, and Yuqiu Qian. 2018. Maximizing social influence for the awareness threshold model. In *DASFAA*. 491–510.

[88] Jing Tang, Xueyan Tang, and Junsong Yuan. 2016. Profit maximization for viral marketing in online social networks. In *ICNP*. 1–10.

[89] Shaojie Tang and Jing Yuan. 2020. Influence maximization with partial feedback. *Operations Research Letters* 48, 1 (2020), 24–28.

[90] Youze Tang, Yanchen Shi, and Xiaokui Xiao. 2015. Influence maximization in near-linear time: A martingale approach. In *SIGMOD*. 1539–1554.

[91] Youze Tang, Xiaokui Xiao, and Yanchen Shi. 2014. Influence maximization: Near-optimal time complexity meets practical efficiency. In *SIGMOD*. 75–86.

[92] Yongxin Tong, Libin Wang, Zimu Zhou, Lei Chen, Bowen Du, and Jieping Ye. 2018. Dynamic pricing in spatial crowdsourcing: A matching-based approach. In *SIGMOD*. 773–788.

[93] FATMANUR AKDOĞAN UZUN, DOĞAN ALTAN, Ercan Peker, MAHMUT ALTUĞ ÜSTÜN, and Sanem Sariel. 2020. Optimization of real-world outdoor campaign allocations. *Turkish Journal of Electrical Engineering and Computer Sciences* 28, 3 (2020), 1276–1292.

[94] Qian Wang, Rajan Batta, Joyendu Bhadury, and Christopher M Rump. 2003. Budget constrained location problem with opening and closing of facilities. *Computers & Operations Research* 30, 13 (2003), 2047–2069.

[95] Linlin Wu, Saurabh Kumar Garg, Rajkumar Buyya, Chao Chen, and Steve Versteeg. 2013. Automated SLA negotiation framework for cloud computing. In *International Symposium on Cluster, Cloud, and Grid Computing*. 235–244.

[96] Xiaomin Xi, Ramteen Sioshansi, and Vincenzo Marano. 2013. Simulation–optimization model for location of a public electric vehicle charging infrastructure. *Transportation Research Part D: Transport and Environment* 22 (2013), 60–69.

[97] Yipeng Zhang, Yuchen Li, Zhifeng Bao, Songsong Mo, and Ping Zhang. 2019. Optimizing Impression Counts for Outdoor Advertising. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, Ankur Teredesai, Vipin Kumar, Ying Li, Rómer Rosales, Evimaria Terzi, and George Karypis (Eds.). ACM, 1205–1215.

[98] Yipeng Zhang, Yuchen Li, Zhifeng Bao, Baihua Zheng, and HV Jagadish. 2021. Minimizing the Regret of an Influence Provider. In *SIGMOD*. 2115–2127.

[99] Yuqing Zhu, Jing Tang, and Xueyan Tang. 2020. Pricing influential nodes in online social networks. *Proceedings of the VLDB Endowment* 13, 10 (2020), 1614–1627.