Social Influencer Selection by Budgeted Portfolio Optimization

Ricardo José López-Dawn, Anastasios Giovanidis

Social Influencer Selection by Budgeted Portfolio Optimization

Ricardo José López Dawn
Sorbonne University, CNRS-LIP6
F-75005, Paris, France
Ricardo.Lopez-Dawn@lip6.fr

Anastasios Giovanidis
Sorbonne University, CNRS-LIP6
F-75005, Paris, France
Anastasios.Giovanidis@lip6.fr

***Abstract*—Influencer marketing has become in the recent years a thriving industry that includes more than 1120 agencies worldwide and with a global market value expected to reach 15 billion dollars by 2022. The advertising problem that such agencies face is the following: given a monetary budget find a set of appropriate influencers on a social platform and recruit them to create a number of posts for the promotion of a certain product. The objective of the campaign is to maximize some impact metric, e.g. the number of impressions, the sales, or the audience reach. In this work, we present an original formulation of the budgeted campaign orchestration problem as a convex program, and further derive a near-optimal algorithm to solve it efficiently. The proposed algorithm has low computational complexity and can scale well for problems with large numbers (millions) of social users, encountered in real-world platforms. We apply our algorithm to a Twitter data set and illustrate the optimal campaign performance for various metrics of interest.**

I. Introduction

In the age of Online Social Platforms (OSPs)[1], audiences trust contents generated by influential users more than advertisement promoted by the platform itself [1]. This can be observed in the social sphere, where brands like Puma [2], advertised their #IgniteXT line through posts created by 61 influencers to promote their line among young public. Another well-known example was Spotify with its #thatsongwhen influencer campaign [3], where, the objective was to increase its new subscribers through posts created by influencers in its favour. Other examples are documented and can be found in different industries as Mattel, Dreamworks, Netflix, etc.

At present, there exist more than 1120 influencer marketing focused agencies, the average earned media value per $1 spent has increased to $5.78 [4], and it is estimated that the influencer marketing industry is on track to be worth up to $15 billion dollars by 2022. Due to the growing relevance and magnitude of this market, a general framework is necessary to determine how to select influencers to create posts in favour of some company that can maximize the objective of the advertising campaign subject to a monetary budget constraint.

The great majority of the objectives in an advertising campaign aims to maximize one of the following metrics [4]:

1) *Impressions*: The total number of times that the content related to the campaign has been displayed in the Newsfeeds of all users on the OSP.
2) *Reach*: The total number of different users that found a post related to the campaign in their Newsfeeds. Impressions and reach are ways to quantify the spread of a campaign in an OSP.
3) *Engagements*: These include the total number of likes, comments and re-posts related to the campaign. This metric captures the interactions received in an advertising campaign.
4) *Conversion/Sales*: Generally, these metrics quantify the ROI (return on investment) which equals the value received from content shared by an advertising campaign.

Influencers can be divided into three categories, neither exclusive nor exhaustive, based on their dissemination capacity:

- *Nano-influencers*: These possess small, niche, and highly engaged audience. Nano-influencers have the smallest number of followers, the highest engagement per post, biggest ROI, and they are easier to recruit.
- *Micro-influencers*: These have the characteristics of being strongly connected with their audience, they tend to receive a lot of engagements per post and are cost-accessible to businesses of all sizes.
- *Macro-influencers*: These share characteristics such as greater reach than micro-influencers, they have a significantly higher cost per post than the micro-influencers, a wide audience and a higher level of professionalism.

Generally the price per post of influencers varies depending on various characteristics like: the type of the social media platform, the number of followers, the average number of engagements per post, the advertised product, etc. Therefore, nowadays we even have companies that are dedicated to the task of how to price the influential users.

Given a monetary budget over a time period where the campaign is deployed, small and medium-sized companies will search for a basket of influencers to maximize their campaign objective (impressions, engagements, reach). Note here that most influencers will not sell all their posting activity for the promotion of a single company/product, so that they can preserve their personal style and offer variety in posting that keeps on feeding their followers' interest.

[1]This work is funded by the ANR (French National Agency of Research) by the "FairEngine" project under grant ANR-19-CE25-0011.

A. Related Literature

The most relevant literature about our Budgeted Portfolio Optimization Problem in OSPs is related to the social influence maximization problem introduced by Kempe et al. in [5]. The elements of this problem are:

- A graph of the social network with the users as the vertex set and social ties among the users as the edge set.
- A diffusion process describing how content is diffused among social neighbors over discrete steps.

In this context the influencer selection problem is stated as follows: for a given size k, choose at most k users of the social network called the seed set, such that the number of users influenced (reached) is maximized when the diffusion process is over [5]. Subsequently, user costs and budgetary restrictions have been introduced [6], [7].

These works result in an NP-hard problem with sub-modular structure that can be sub-optimally solved in polynomial time using greedy approximation algorithms. However, this formulation does not model reality because an influencer does not necessarily attribute his whole activity to the advertising campaign, so it is not sufficient to make a binary decision to include an influencer in the seed set or not. Also, the cost is in reality calculated per post or content produced rather than per recruited user. Furthermore, the knowledge of a model for the dissemination of information in the OSPs is generally not available. Finally, information about the post impressions and engagements can be measured and collected, so there are available data sets available that track the campaign results and the detailed mutual influence between social users.

B. Our Contribution

In this work, we introduce a new formulation of the budgeted portfolio optimization problem of OSPs, which aims to find the *participation ratio* of each user in the campaign acquired by the advertiser that maximizes the campaign objective under budget restrictions. Our formulation takes advantage of the known user activity over a time period, the cost per post of each influencer and the information availability over previously collected data about impressions and user interactions in general. The participation ratio per user is the proportion of user generated posts in favour of the campaign during its realisation. The main differences between our model and Kempe's approach are summarized in the Table I.

To further elaborate on the differences in Table I, in [5] the work concerns the spread of a single post, the knowledge of the diffusion process is necessary and the user selection is binary. On the other hand, in our model the spread of influence is achieved by posting over time, the knowledge of the number of impressions from each source to any other user Newsfeed should be known, and we search for a continuous rate per user.

The formulation of the budgeted portfolio optimization problem of OSPs as well as some further assumptions are provided in Section II. In Section III, we develop two algorithms, one for linear and another for concave objective functions that can solve three particular cases of campaign objective:

Influence Maximization [5]	Our Budgeted Portfolio Optimization
Discrete	Continuous
Graph	User set
Diffusion process	Data set of Impressions
Cost per user	Cost per post
Objective:	Objective:
Maximize the number of users influenced when the diffusion process is over	Maximize the campaign objective (Impressions, Conversion/Sales or Reach)
Return:	Return:
Seed set	Participation ratio per user

TABLE I
DIFFERENCES BETWEEN APPROACHES

Impressions/Engagements: This case arrives when we consider the advertiser's campaign objective as linear. The optimal solution can be found by a greedy algorithm with computational complexity of order $\mathcal{O}(max((N-1)\ log(N-1), D))$ where $N-1$ is the number of users minus the advertiser and $D \leq (N-1)^2$ is the total number of pairs of users who create content and appear in other users' Newsfeeds/Walls. Hence the solution scales well with the number of users.

Conversion/Sales: Under the assumption that the purchasing propensity of users (or the ROI) varies depending on their exposure to product related content, we study campaign objectives that are monotone increasing and concave with respect to user impressions i.e. functions that exhibit diminishing returns. To achieve near-optimal solution, we propose an iterative greedy algorithm with complexity $\mathcal{O}(max((N-1)\ log(N-1), D))$ per iteration. For illustration, we work with the α-fairness utility family. A special case is proportional fairness with sum of logarithmic utility functions as objective.

Reach: Another special case of α-fairness is when α tends to infinity, which gives a Max-Min fairness solution. We can maximize the *Reach* by selecting such specific utility because its solution maximises the number of selected influencers who receive non-zero participation ratio.

In Section IV, the algorithm performance is evaluated on a real data-trace from Twitter and finally, conclusions are drawn in section V. The code is available in [12].

II. The Budgeted Portfolio Optimization Problem

Let us first describe a generic social network platform, such as Facebook or Twitter. A set of users generate and share some content, denoted as posts, through the platform. Each user has a list of followers and a list of leaders. A user can simultaneously be follower and/or leader of others. As a follower, he (she) is interested in the content posted by his (her) leaders. With each user a Newsfeed is associated, which is a list of received posts.

At each specific point in time, a user sees in his (her) Newsfeed posts originated by other users who may or may not be their direct leaders, the number of these posts seen represents the *impressions* on him (her) and the *impression ratio* is the ratio of the impressions originated by some given user over all impressions in a given snapshot. The average

ratio over several snapshots is called the *average impression ratio* in the time window.

We consider a constant number N of active users in a specific time window, forming the set $\mathcal{N}$. Users are labelled by an index $n = 1, ..., N$. We denote by $\lambda^{(n)}$ [posts/time window] the rate with which user n generates new posts, and we make the assumption that content posted instantaneously appears on the Newsfeeds of his followers and is further propagated through the social network. For all users $n \in \mathcal{N}$ we suppose that they preserve their post rate $\lambda^{(n)}$ constant in the time-window.

Let us denote by $p_n^{(j)}$ the average impression ratio of posts that originate from user n in the Newsfeed of user j. This quantity $p_n^{(j)}$ is assumed known for the rest of the article and can be measured or estimated in two ways: *Empirically*, by taking multiple Newsfeed snapshots in the time window and calculating the average of the ratio of impressions between pairs of users over those time points. Alternatively, *through Markovian analysis*. If we have complete knowledge of the social graph and user posting activity, we can derive $p_n^{(j)}$ using the Markovian diffusion model introduced in [8].

Naturally, our average impression ratios satisfy:

$$\sum_{n \in \mathcal{N}} p_n^{(j)} = 1, \ \forall j \in \mathcal{N}. \tag{1}$$

Our model does not require explicit knowledge of the list of followers and leaders of each user, nor a diffusion process as in the approach by Kempe et al. [5]. However, it does require knowledge over the average impression ratio, that contains all this information resulting from diffusion. Furthermore, we are interested in studying the relative impact between pairs of users and not the absolute impact, since the Walls and Newsfeeds can vary in size between users.

Note here that in Instagram and other OSPs, due to the lack of a re-posting option the propagation of information is only given to the immediate followers of a user, thus hindering post-propagation. These networks are simpler to describe; the user sets form a bipartite graph (leaders/followers).

A. The budgeted portfolio optimization problem

In the budgeted portfolio optimization problem an advertiser $i \in \mathcal{N}$ with a certain monetary budget B [EUR/time window] in his (her) disposal orchestrates an advertising campaign in a unit of time (equal to the time window) by investing on other users to create posts in his (her) favour. The aim is to maximize some impact metric, e.g. the number of impressions, the sales, or the audience reach.

We suppose that for each user $n \neq i$ there is an associated cost per post c_n [EUR/post] so that the user n will be willing to create posts in favor of the advertiser i.

In order to formulate this optimisation problem, we need to quantify the participation of each user n in the campaign of the advertiser i. Hence, we define for each user $n \neq i$, the *continuous participation ratio* $a_n \in [0, 1]$ in the campaign as the unknown proportion of user n's generated posts acquired by the advertiser i in the unit of time. We fix $a_i = 1$ meaning that the advertiser always posts to promote its own product. Then, $a_n \lambda_n$ [posts/time window] represents the number of posts that the user n creates in favor of the advertiser i.

Similarly, we define by $p_n^{(j)}(a_n)$ the *campaign-related impression ratio* as the average value of the impression ratio in the Newsfeed of user j originating from user n and related to the campaign of the advertiser i. The campaign-related impression ratio can be similarly estimated and measured as above and satisfies:

$$p_n^{(j)}(a_n) \leq p_n^{(j)} \quad \forall n, j \in \mathcal{N}. \tag{2}$$

The empirical probability that an impression reaching user j is related to the campaign is called the *potential of user j*:

$$\omega^{(j)}(\mathbf{a}) = \sum_{n \in \mathcal{N} \setminus \{j\}} p_n^{(j)}(a_n) \leq 1. \tag{3}$$

In the above $\mathbf{a} = (a_1, ..., a_N)^T$ is the participation vector of all the users into the advertising campaigns of user i.

We introduce a utility function U_j for each user j that maps the potential of user j, $\omega^{(j)}$, to the campaign objective of the advertiser i. Different expressions for U_j model different performance metrics.

The budget invested to user $n \neq i$ by the advertiser i is $B_n(a_n) = c_n a_n \lambda^{(n)}$ [EUR/time window] and the total budget of the advertiser i is B [EUR/time window]. Therefore the constraints in our budgeted portfolio optimization problem will be naturally a budget restriction $\sum_{n \neq i} B_n(a_n) \leq B$ and the continuous unknown variables $a_n \in [0, 1]$. Alltogether, we can formulate the general budgeted portfolio optimization problem

$$\begin{aligned} \max_{\{a_n\}_{n \neq i}} \quad & \sum_{j \in \mathcal{N} \setminus \{i\}} U_j(\omega^{(j)}(\mathbf{a})), \\ \text{s.t.} \quad & \sum_{n \in \mathcal{N} \setminus \{i\}} c_n a_n \lambda^{(n)} \leq B, \\ & a_i = 1, \ 0 \leq a_n \leq 1, \forall n \in \mathcal{N}. \end{aligned} \tag{BPO}$$

B. Variations and extensions

The above formulation allows us to to introduce further extensions of our model:

1) We can consider that certain users want to sell no more than a certain ratio of their posts $a_n \leq r_n \leq 1, \forall n$.
2) Another variation is by introducing a set of posting categories to every user ς_n and to activate an influencer-audience member pair (n, j), only when the two users share some common interests. In this case the potential of influencer n is expressed as:

$$\omega^{(j)}(\mathbf{a}) = \sum_{n \in \mathcal{N} \setminus \{j\}} p_n^{(j)}(a_n) I_{\varsigma_n \cap \varsigma_j \neq \varnothing},$$

with $\varsigma_n \subset \{1, ..., \text{Number of categories}\}$ the hobbies or interests of user n and similarly for ς_j.

C. Assumption on ad propagation and impact metrics

An assumption for the rest of the article is that we consider a linear propagation for the posts related to the campaign and seen on the Newsfeeds, namely:

$$p_n^{(j)}(a_n) = a_n p_n^{(j)}. \tag{4}$$

This is reasonable because if the user j is an immediate follower of influencer n, and all posts from the influencer appear on his Newsfeed, then a percentage a_n will be related to the campaign. This is actually the case for platforms without sharing, like Instagram, but for other platforms, impressions could arrive through sharing of content from intermediate users. Then, the above linear expression implies that a post from j is shared randomly, independent of its content, which of course is not true. We will use however the linear assumption as a reasonable approximation to the campaign diffusion process for any platform, because we lack of any prior information related to how users might react to the campaign's posts. So, for the rest of the article, the potential of the user j is expressed as:

$$\omega^{(j)}(\mathbf{a}) = \sum_{n \in \mathcal{N} \setminus \{j\}} a_n p_n^{(j)}. \tag{5}$$

The utility function U_j in [BPO] of the user j, represents from a modeling point of view the following:

- *Impressions/Engagements*: In this case, the objective function for each user is a linear function. This translates as follows: an increase in the impression potential (5) of a user j results in a proportional increase in their utility.
- *Conversion/Sales*: The α-fairness utility function models diminishing returns over the potential of each user j. As the amount of one participation ratio increases, then after some point the marginal conversion/sales (extra output gained by adding an extra unit) decreases. We will use the logarithmic function in particular to measure Sales.
- *Reach*: We model this case by applying user-specific thresholds ϵ_j for each user j. If the user j sees more than the threshold ϵ_j campaign-related impression ratio $\omega^{(j)}(\mathbf{a})$, then the user j is consider to be reached by the campaign:

$$U_j(\omega^{(j)}(\mathbf{a})) = I_{\omega^{(j)}(\mathbf{a}) > \epsilon_j}. \tag{6}$$

Hence, under the assumption of a linear propagation model and activity constraints $\{r_n\}_{n \neq i}$ we have the formulation of the budgeted portfolio optimization problem in OSPs for various objectives (corresponding to impact metrics):

$$\begin{aligned} \max_{\{a_n\}_{n \neq i}} \quad & \sum_{j \in \mathcal{N} \setminus \{i\}} U_j\Big(\sum_{n \in \mathcal{N} \setminus \{j\}} [a_n p_n^{(j)}] \Big), \\ \text{s.t.} \quad & \sum_{n \in \mathcal{N} \setminus \{i\}} c_n a_n \lambda^{(n)} \leq B, \qquad \text{[BPO-G]} \\ & a_i = r_i, \ 0 \leq a_n \leq r_n, \forall n \in \mathcal{N}. \end{aligned}$$

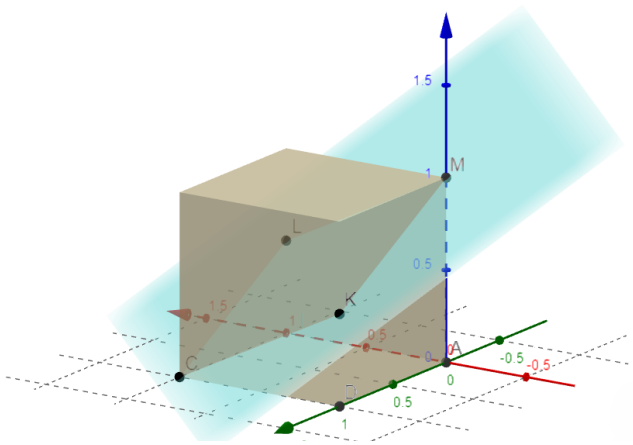

Fig. 1. Example of extreme points for a specific campaign.

D. Feasibility set

For increasing utility functions the budget constraint is satisfied with equality at the optimal solution. So, the feasible set S in [BPO-G] is reduced to:

$$\begin{aligned} S = \{(a_n)_{n \neq i} \in \mathbb{R}^{N-1} : & \sum_{n \in \mathcal{N} \setminus \{i\}} c_n a_n \lambda^{(n)} = B, \\ & \forall n \in \mathcal{N} \setminus \{i\}, \ 0 \leq a_n \leq r_n\}. \end{aligned} \tag{7}$$

Note that the feasible set has dimension $N-1$, excluding the advertiser who always gets $a_i^* = 1$. We denote by $E(S)$ the set of extreme points of S.

In Figure 1, we show one example of a campaign in a network of four users with advertiser user $\#4$ (not included in the figure since always $a_{\#4}^* = 1$) accompanied by a unitary budget $B = 1$ and three users $\#1, 2, 3$ with unitary activity restrictions $a_n \in [0, 1]$ and total costs $(c_1\lambda^{(1)}, c_2\lambda^{(2)}, c_3\lambda^{(3)}) = (\frac{1}{2}, \frac{1}{2}, 1)$ respectively. Hence, the set S is the blue plane segmenting the brown cube and the extreme points $E(S)$ are the points $M = (0, 0, 1), L = (1, 0, \frac{1}{2}), C = (1, 1, 0)$, and $K = (0, 1, \frac{1}{2})$. Observe that these points (allocation vectors) have coordinates with specific structure: for each point, all its coordinates are binary $\{0, 1\}$ except at most one entry which can be in $(0, 1)$. This observation is a general attribute of $E(S)$ as we prove below.

Proposition 1. *(i) Let $(a_n)_{n \neq i} \in E(S)$ be an extreme point of S in [BPO-G], then it satisfies the next property: there exists at most one $j \in \mathcal{N} \setminus \{i\}$ such that $a_j \in (0, r_j)$ and $\forall l \in \mathcal{N} \setminus \{i, j\}$, $a_l \in \{0, r_l\}$. Conversely, any point $\{a_n\}_{n \neq i} \in S$ that satisfies this property is an extreme point. (ii) Furthermore, a global maximizer in [BPO-G] can be written as a convex combination of points satisfying this attribute.*

Proof. (i) The feasible set S is a convex polytope, defined by the intersection of a hyper-plane (from the budget equality) with an $(N-1)$-dimensional hyper-cube (from the range of participation ratios). Each extreme point of S, $(a_n)_{n \neq i} \in E(S)$ lies on an edge (or vertex) of the hyper-cube. The points at each edge of the cube have all dimensions either

0 or 1 except one dimension varying in $[0,1]$. So the extreme point of S will also share this property. Conversely, consider a point $(a_n)_{n\neq i} \in S$ satisfying this property. Such point cannot be written as a convex combination of two other points $s_1, s_2 \in S$: to see this $\theta s_{1,n} + (1-\theta)s_{2,n} = a_n$, for $n \neq i$, and $\theta \in (0,1)$. Then, $a_n = 1 \Rightarrow s_{1,n} = s_{2,n} = 1$ and $a_{n'} = 0 \Rightarrow s_{1,n'} = s_{2,n'} = 0$, hence all coordinates of points $s_1, s_2 \in S$ are equal to those of $(a_n)_{n\neq i} \in S$, except in dimension j. This means that $s_1, s_2, (a_n)$ lie on the same edge of the hyper-cube, which is impossible by construction.

(ii) A global maximizer of [BPO-G] is necessarily in S. A point in S can be written as a convex combination of the extreme points $E(S)$, because S is a compact convex subset of $\mathbb{R}^{N-1}$, and therefore S can be expressed as the convex hull of its extreme points [11]. □

We can explicitly determine the extreme points in the set $E(S)$ of the feasible set S as follows: for every permutation $\{i_n\}$ of the user set, we pick sequentially one user after the other in order and allocate full budget $a_{i_n} = r_{i_n}$ to the i_n-th user based on the permutation order, until the budget B is completely consumed. Hence, only the last allocated user i_s can get allocated a ratio $a_{i_s} \in (0,1)$. All user with indices larger than i_s will get 0 ratio.

III. Solution to the advertiser's campaign

In this section we present the optimization problem, for linear and concave utility functions and in each case, we present specific algorithms to solve [BPO-G]. The above properties of the feasible set and Proposition 1 will be used here to propose low-complexity fast algorithms for both cases. Notation and parameters are summarized in Table II:

Budgeted Portfolio Optimization Problem	
User set	$\mathcal{N}$
Advertiser	$i \in \mathcal{N}$
Average impression ratios	$\{p_n^{(j)}\}_{n,j\in\mathcal{N}}$
Cost per post	$\{c_n\}_{n\in\mathcal{N}}$
Budget over a time period	B
Activity restriction (optional)	$\{r_n\}_{n\in\mathcal{N}}$
User activity	$\{\lambda^{(n)}\}_{n\in\mathcal{N}}$
Objective:	
Campaign objective	$\sum_{j\in\mathcal{N}\setminus\{i\}} U_j(\sum_{n\in\mathcal{N}\setminus\{j\}}[a_n p_n^{(j)}])$
Return:	
Participation ratio per user	$\{a_n\}_{n\neq i}$

TABLE II
Elements of our problem and notation

A. Linear utility function

In this case, $U_j(\omega^{(j)}(\mathbf{a})) = \omega^{(j)}(\mathbf{a})$. Therefore, solving this problem is equivalent to maximizing the Impressions/Engagements as campaign objective. By defining $\phi_n = \sum_{j\in\mathcal{N}\setminus\{n,i\}} p_n^{(j)}, \forall n \in \mathcal{N}\setminus\{i\}$, and $\phi_i = \sum_{j\in\mathcal{N}\setminus\{i\}} p_i^{(j)}$, we express our objective function as:

$$\sum_{j\in\mathcal{N}\setminus\{i\}} \Big(\sum_{n\in\mathcal{N}\setminus\{j\}} [p_n^{(j)} a_n] \Big) = \sum_{n\in\mathcal{N}\setminus\{i\}} (a_n\phi_n) + a_i\phi_i.$$

The global optimum of the linear optimization problem is an extreme point $(a_n^*) \in E(S)$ and can be found by a greedy algorithm as follows:

The users $\{i_k\}_{k=1,\dots,N-1}$ are indexed by decreasing order by their ϕ_{i_k} per EUR, $\{\frac{\phi_n}{c_n\lambda^{(n)}}\}_{n\neq i}$. Hence, the user i_1 generates the largest cumulative impressions per EUR, the user i_k generates the k-th maximal number of cumulative impressions per EUR and so on. Define the marginal budget of user i_l as $B_l = B - \sum_{k<l} a_{i_k} c_{i_k} \lambda^{(i_k)}$, $B_1 = B$, and let us define $\forall l \in \{1, ..., N-1\}$:

$$a_{i_l}^* = r_{i_l} I_{c_{i_l} r_{i_l} \lambda^{(i_l)} \leq B_l} + \frac{B_l}{c_{i_l}\lambda^{(i_l)}} I_{c_{i_l} r_{i_l} \lambda^{(i_l)} > B_l}.$$

Then, by construction $\sum_{l<N} a_{i_l}^* c_{i_l} \lambda^{(i_l)} \leq B$ and $\{a_n^*\}_{n\neq i}$ is an optimal vector for our portfolio optimization problem and an extreme point in $E(S)$.

Note that if $\exists n_1, n_2 \neq i$ with $\frac{\phi_{n_1}}{c_{n_1}\lambda^{(n_1)}} = \frac{\phi_{n_2}}{c_{n_2}\lambda^{(n_2)}}$, then our optimum may not be the only optimum, but it is unique modulo permutations of the set with equal elements $\frac{\phi_n}{c_n\lambda^{(n)}}$.

Notice that this algorithm has a computational complexity of order $\mathcal{O}(max((N-1)\, log(N-1), D))$ where D is the number of non-zero average impression ratios between pairs of users (using merge sort to order the set $\{\frac{\phi_n}{c_n\lambda^{(n)}}\}_{n\neq i}$). Therefore, it is a good algorithm to use in large data sets.

B. Concave utility function

In this subsection, we solve the general case [BPO-G]. Before moving on to the general algorithmic solution, we mention that as particular choices of concave utility functions, we could consider the α-fairness family of utility functions [9], [10]. This is a general class of utility functions that captures different fairness criteria such as proportional fairness for $\alpha \to 1$ ($U_j(\omega^{(j)}) = \log(\omega^{(j)})$) and max-min fairness for $\alpha \to \infty$. It also captures many other fairness criteria that lie between them with a suitable choice of the parameter $\alpha \in (0,\infty) \setminus \{1\}$.

$$U_j(\omega^{(j)}) = \gamma_j \frac{(1+\omega^{(j)})^{1-\alpha}}{1-\alpha}. \tag{8}$$

Here, $\gamma_j \in \mathbb{R}$ is a given weight that we will assume unitary in the absence of information. From a modeling perspective, the case $\alpha \to 1$ can maximise Sales, if these are modeled as a logarithmic function of impressions, and the case $\alpha \to \infty$ can maximise Reach, i.e. the users who can receive an impression related to the campaign.

Note that U_j is strictly increasing in $\omega^{(j)}(\mathbf{a})$, then [BPO-G] is equivalent to:

$$\begin{aligned} \max_{\{a_n\}_{n\neq i}} \quad & \sum_{j\in\mathcal{N}\setminus\{i\}} U_j(\omega^{(j)}), \\ \text{s.t.} \quad & 0 \leq \omega^{(j)} \leq \sum_{n\in\mathcal{N}\setminus\{j\}} [a_n p_n^{(j)}],\ \forall j \in \mathcal{N}\setminus\{i\}, \\ & \sum_{n\in\mathcal{N}\setminus\{i\}} (c_n a_n \lambda^{(n)}) - B = 0, \\ & a_i = r_i,\ 0 \leq a_n \leq r_n, \forall n \in \mathcal{N}, \end{aligned}$$

where $\omega^{(j)}$ is an auxiliary variable.

It is common in convex optimization problems to use primal-dual approaches, however since we are interested in developing algorithms for real-world platforms with millions of variables (one variable per user) and millions of constraints (one constraint per user), then the number of primal and dual variables will be enormous and the convergence of primal-dual algorithms with such sizes is problematic. Hence, we need to appeal to heuristic solutions that take advantage of the structure of the feasibility set and can well approximate the optimum for very large size of N. The main idea of our algorithm is to greedily select per step an extreme point of S, that maximises the improvement in the objective function and average over all previous selected points.

First we write the Lagrangian function only with respect to the auxiliary $\{\omega^{(j)}\}_{j\neq i}$ and B budget constraints.

$$\mathcal{L}(\mathbf{a},\omega;\nu,\mu) = \sum_{j\in\mathcal{N}\setminus\{i\}} U_j(\omega^{(j)}) + \nu(B - \sum_{n\in\mathcal{N}\setminus\{i\}} c_n a_n \lambda^{(n)}) + \sum_{j\in\mathcal{N}\setminus\{i\}} \mu_j(\sum_{n\in\mathcal{N}\setminus\{j\}} a_n p_n^{(j)} - \omega^{(j)}),$$

We can apply the KKT conditions and solve the primal problem. We get for the $\omega^{(j)}$ primal variables and any pair of duals (ν,μ):

$$\frac{\partial\mathcal{L}}{\partial\omega^{(j)}} = U_j'(\omega^{(j)}) - \mu_j = 0. \tag{9}$$

So, since $U_j'(\omega^{(j)}) > 0$ for all $\omega^{(j)}$, we get at the optimum:

$$\mu_j^* = U_j'(\omega^{(j)*}). \tag{10}$$

On the other hand, for the a_n primal variables:

$$\frac{\partial\mathcal{L}}{\partial a_n} = -\nu c_n\lambda_n + \sum_{j\in\mathcal{N}\setminus\{i,n\}} \mu_j p_n^{(j)}. \tag{11}$$

Observe that since the activity restrictions $[0, r_n]$ are not considered in the Lagrangian, then we cannot set (11) equal to 0. There are three cases for (11) given a dual pair (ν,μ):

(i) $\frac{\partial\mathcal{L}}{\partial a_n}(\nu,\mu) > 0$, implies that $a_n^* = r_n$ because the maximum is found for the largest value of a_n.

(ii) $\frac{\partial\mathcal{L}}{\partial a_n}(\nu,\mu) < 0$, implies that $a_n^* = 0$ because the maximum is found for the smallest value of a_n.

(iii) $\frac{\partial\mathcal{L}}{\partial a_n}(\nu,\mu) = 0$, implies $a_n^* \in (0, r_n)$.

The above holds also for the optimal dual pair. Hence, we see that given the optimal threshold ν^* and the optimal prices μ^*, we can decide whether some a_j^* is 0, or r_j, or in-between. From the case (iii) and by (10), we observe for some $k \in \mathcal{N}\setminus\{i\}$ that $\frac{\partial\mathcal{L}}{\partial a_k}(\nu^*,\mu^*) = 0$ if and only if:

$$\nu^* = \frac{1}{c_k\lambda_k} \sum_{j\in\mathcal{N}\setminus\{i,k\}} U_j'(\omega^{(j)*}) p_k^{(j)} := Q_k^*, \tag{12}$$

Altogether, the conditions (10) and (11.i-.iii) along with (12) summarize the KKT conditions.

Hence, let us proceed to give an iterative *greedy* heuristic by using the KKT conditions and the fact that the global optimum can be written as a convex combination of the extreme points by Proposition 1. For this purpose, we introduce the vector Δ (to be explained latter) which is initialised at step $t = 0$ as $\Delta(0) = 0_{N-1}$, and $\mu(0) = (\mu_n(0))_{n\neq i}$, assigning very large values. Large μ_j corresponds to a very small $\omega^{(j)}$ close to 0 by (10) and the property of diminishing returns for the utility function U_j. We update at step $t+1$ as follows:

- Step A: Update extreme points. With the $\mu(t)$ vector, we calculate for all users $n \in \mathcal{N}\setminus\{i\}$:

$$Q_n(t+1) = \frac{\sum_{j\in\mathcal{N}\setminus\{i,n\}} \mu_j(t) p_n^{(j)}}{\lambda_n c_n} \tag{13}$$

The threshold ν^* splits (refer to (11.i,.ii,.iii)) the user set into those users who get zero participation, those who get maximum, and those with $Q_n^* = \nu^*$, from (12). Then we proceed to order the Q's in decreasing order of value and we store the user indices $Per(t+1) = \{i_1, i_2, ..., i_{N-1}\}$ given by the order at step $t+1$. The users with highest $Q(t+1)$ will be much higher than ν^* (still unknown). So, let us first choose as participation ratio at step $t+1$:

$$a_{i_1}(t+1) = r_{i_1}, ..., a_{i_s}(t+1) = r_{i_s},$$

$$a_{i_{s+1}}(t+1) = B - \sum_{j=1}^{s} c_{i_j}\lambda_{i_j} r_{i_j}, \quad \& \quad a_{i_{s+2}}(t+1) = 0, ...$$

i.e. we allocate greedily the budget to the users with highest Q, while satisfying the KKT condition (11.i,.ii,.iii). By construction at each step $a(t+1) \in E(S)$ is an extreme point of S, see Proposition 1. Note that $a(t+1) = (a_n(t+1))_{n\neq i}$, where n are the original indices.

- Step B: Update Averages. We update $\Delta(t+1)$ using the extreme points obtained until step $t+1$ as:

$$\Delta(t+1) = \frac{t\ \Delta(t)}{t+1} + \frac{1}{t+1}a(t) = \frac{1}{t+1}[a(1) + ... + a(t+1)],$$

where $\Delta(t+1)$ represents the average of the extreme points found throughout the process, and certainly $\Delta(t+1) \in S$ because it is a convex combination of the extreme points which are in S. Intuitively, the algorithm successively over t selects an extreme point based on the KKT conditions in (11.i,.ii.iii) and averages it over the previously selected extreme points.

- Step C: Update ω and μ. For each user $j \in \mathcal{N}\setminus\{i\}$ we calculate $\omega^{(j)}(t+1)$ and $\mu_j(t+1)$ in function of the average $\Delta(t+1)$ using the definition (3) and the KKT condition (10) respectively:

$$\omega^{(j)}(t+1) = \sum_{n\in\mathcal{N}\setminus\{j\}} \Delta_n(t+1) p_n^{(j)}, \tag{14}$$

$$\mu_j(t+1) = U_j'(\omega^{(j)}(t+1)). \tag{15}$$

- Step D: Stopping criterion. We return as output association $a^* = \Delta(t+1)$ when $||\Delta(t+1) - \Delta(t)|| < \epsilon$ or $||\sum_{j\in\mathcal{N}\setminus\{i\}} U_j(\Delta(t+1)) - \sum_{j\in\mathcal{N}\setminus\{i\}} U_j(\Delta(t))|| < \epsilon$ or when the maximum number of iterations T is attained.

Notice that at the step $t+1$, (14) and (15) guarantees us that the condition (9) of the KKT condition is satisfied and the condition (11) is satisfied because the budgets are allocated to the users following a threshold policy, albeit the value of ν^* is unknown. Note in turn that we artificially introduce the average of the extreme points because otherwise the selection of extreme points alone would be unstable and oscillate, whereas it would not allow for a selection of any point inside the feasible set S.

Let us note that our algorithm is greedy, in the sense that it successively chooses per step the maximum direction of growth of the Lagrangian by applying (15) and the allocation in Step A, which follows (11.i,.ii,.iii). Namely, we are proposing a greedy gradient method on the set of extreme points. Due to the use of average update of $\Delta(t+1)$ in Step B our algorithm will eventually converge. The convergence to the optimum is not garanteed, but numerical evaluations show that our greedy approach that respects the KKT conditions stepwise performs sufficiently well. Each iteration has computational complexity of order $\mathcal{O}(max((N-1)\ log(N-1), D))$ and the algorithm converges *sub-linearly* in a finite number of steps.

In the linear utility case, the greedy algorithm reduces to the one proposed in the previous sub-section and runs in a single iteration with the same computational complexity.

IV. Numerical evaluations

The aim of this section is to evaluate the performance of our algorithms for various campaign objectives using information from a real large Twitter data trace [13]. This database represents the activity of users on Twitter during the 2018 Russian elections. In particular for our purposes, we use a 4-uple per post with the following information obtained from this database: $[TweetID,\ TimeStamp,\ UserID,\ RetweetID]$.

The dataset is described as a list of such 4-uples. Each participating user and Tweet have a unique associated UserID and TweetID respectively. RetweetID represents the TweetID which was retweeted (or -1 if it is a self-post) and TimeStamp is the time that the Tweet was (re)-posted. The entire database spans 57 days and involves $181,621$ different $UserID$s. Moreover, there is an average of 3.71 posts, an average of 7 re-posts per user and we find $87,987$ users who have re-posted (shared a post) at least once. These users can be potentially reached by any advertising campaign, since they share content.

From the dataset, we derive the empirical post and re-post rate for every user $\{\lambda^{(n)}\}_{n\in\mathcal{N}}$ and $\{\mu^{(n)}\}_{n\in\mathcal{N}}$ respectively. We can further infer a friendship graph using the relationships of retweets (RetweetID), by drawing a directed edge from leader to follower, each time a user retweets something. We call this a "star" graph due to its shape: it contains $181,621$ nodes, $517,421$ edges with a mean degree of 5.70 followers per user. Among the users, $167,646$ users lack of followers and only $13,975$ users have followers. The latter can be potential influencers and we denote their set by $\mathcal{L}$.

We classify the $13,975$ potential influencers into 3 categories: $7,512$ users have $1-3$ followers and are potentially Nano-influencers; $5,064$ have $4-34$ followers and are potentially Micro-influencers; and $1,399$ have more than 34 followers and are potentially Macro-influencers. Having complete knowledge of the social graph and the posting and re-posting rates, the average impression ratios $\{p_n^{(j)}\}_{n,j\in\mathcal{N}}$ and the average engagement ratios $\{q_n^{(j)}\}_{n,j\in\mathcal{N}}$ can be estimated by the Markovian method introduced in [8] (see Section II.A). By definition, the engagements are the shared impressions during the 57 days. We introduce a constant $\delta := \sum_{n\in\mathcal{L}} \frac{|\mathcal{L}^{(n)}|(\lambda^{(n)}+\mu^{(n)})}{|\mathcal{L}|}$ equal to the average number of impressions in the Newsfeed seen by some user in the network within a unit of time. In this expression $|\mathcal{L}^{(n)}|$ is the number of leaders of user n and $\mathcal{L}$ is the set of users in $\mathcal{N}$ who are leaders of at least one user.

As a next step, we need to determine the cost per post c_n charged to the advertiser user i by user n. On Twitter, it is a common market practice to consider the cost per post of user n as $2\frac{\#Followers_n}{1000}$ [EUR/post]. Since that our database is of the order of 10^5 users and Twitter is of the order of 10^8, we will assume a normalization constant in the number of followers of 10^3, so our cost per post of user n to consider is 2 $\#Followers_n$ [EUR/post].

For the evaluations we will consider no restrictions on user participation ratios ($r_n = 1, \forall n \in \mathcal{N}$) in the absence of information. Finally, we select as advertiser the user with UserID $= 2513730044$, who has 15 $\#Followers$. This user is potentially a Micro-influencer, like many stores that provide services in a certain medium-populated area.

We proceed to solve and to find the optimal solutions $\mathbf{a}_L^*, \mathbf{a}_{Log}^*$ and $\mathbf{a}_M^*$ of [BPO] according to three functions respectively: Linear function by Algorithm 1, Logarithmic function and Max-min function by Algorithm 2. The stopping criterion in Algorithm 2 is when the number of iterations reaches a maximum equal to 30, or when the infinite norm of solutions between iterations is less than 0.1%. Using these solutions $\mathbf{a}^* \in \{\mathbf{a}_L^*, \mathbf{a}_{Log}^*, \mathbf{a}_M^*\}$, we evaluate the metrics:

- *Total number of Impressions*: $\delta\ \sum_{j\in\mathcal{N}\setminus\{i\}} \omega^{(j)}(\mathbf{a}^*)$.
- *Total Sales*: $\sum_{j\in\mathcal{N}\setminus\{i\}} log(\delta\ \omega^{(j)}(\mathbf{a}^*)+1)$.
- *Total Reach*: $\sum_{j\in\mathcal{N}\setminus\{i\}} I_{\omega^{(j)}(\mathbf{a}^*)>\epsilon}$ with ϵ a threshold which denotes when a user has been reached by the campaign. We select two values: $\epsilon = 0$ for the upper reference curve, and $\epsilon = \delta$ obtained as the average rate of impressions in the Newsfeed.
- *Selected number of Nano-, Micro-, and Macro-influencers*: The number of users with $a_j^* \neq 0$. Specifically: those up to 3 followers are Nano- (fewer followers than 60% of users in our database), those with $3-34$ followers are Micro- (more followers than 60% of users but fewer than 90% in our database), and those with more than 34 followers are Macro- (more followers than 90% of users in our database) respectively.

The six plots in Fig. 2 illustrates how the above metrics change with increasing monetary budget per day, for each of the three different campaign objectives (Linear, Logarithmic and Max-min). Note here that Algorithm 2 has sub-linear convergence as empirically observed. More precisely, from

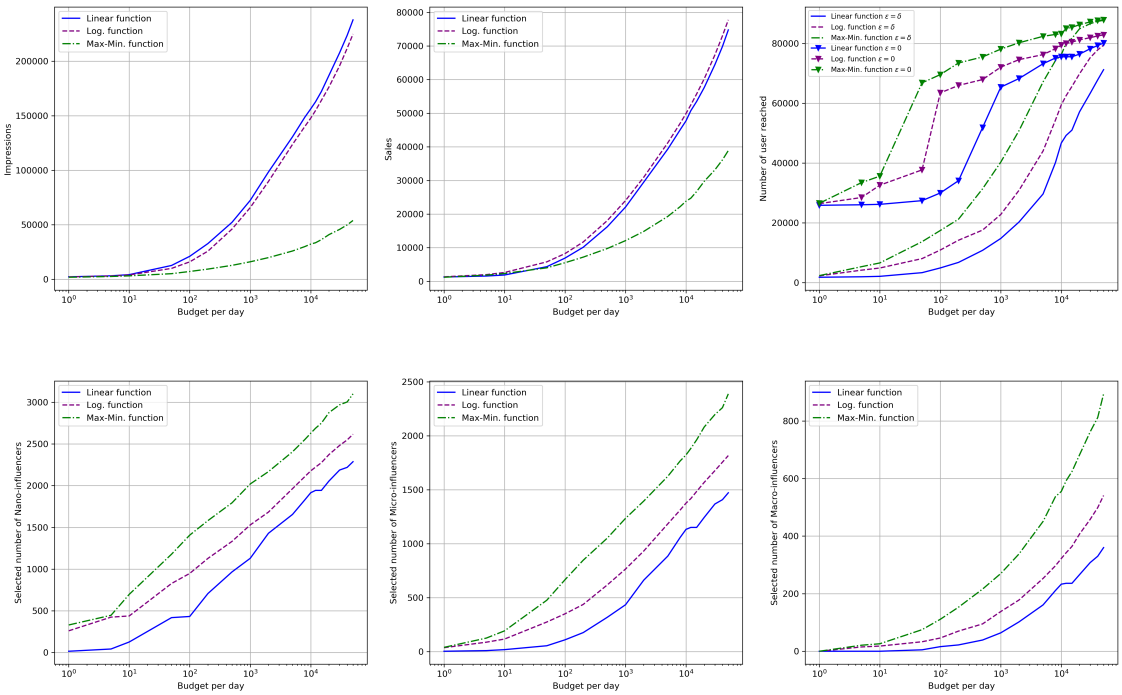

Fig. 2. Metrics across different campaign policies and number of selected influencers across different campaign policies.

Figure 2 we observe the following:

Linear objective - This campaign gives the most impressions performance because this metric coincides with the objective. It gives very high sales for large budgets (more than $50[EUR/day]$), but has the worst reach performance everywhere independent of ϵ. Interestingly, it selects the least number of influencers in all categories, for any budget given.

Logarithmic objective - This campaign gives the best sales performance because this metric coincides with the campaign. Also, very high impressions and a moderate reach performance. It selects more influencers than the linear, in all categories.

Max-min objective - This campaign gives the best audience reach for any given budget and ϵ chosen, but performs bad in Sales and impressions. In fact, for a budget $> 40K[EUR/day]$ the campaign can reach all possible users for both ϵ values.

For all three objectives, the optimal policy selects mostly Nano- and Micro-influencers in low budgets. Macro-influencers are selected for larger budgets. In fact, the number of Nano- and Micro-influencers selected increases logarithmically with increasing $Budget$, whereas the number of Macro-influencers linearly with the $Budget$.

V. Conclusions

In this work, we have presented an original formulation of the budgeted campaign orchestration problem to maximize some impact metric. We have derived a convex program and then a near-optimal algorithm to solve it efficiently. This algorithm has low computational complexity and can scale well for problems with large numbers (millions) of social users, encountered in real-world platforms. We have applied our algorithm to a Twitter data set and illustrate the optimal campaign performance for various metrics of interest.

References

[1] MacKinnon, K.A.. User Generated Content vs. Advertising: Do Consumers Trust the Word of Others Over Advertisers?. *The Elon Journal of Undergraduate Research in Communications*, Vol. 3, No. 1, 2012.

[2] *https://www.brandmanic.com/puma-influencers/*

[3] *https://spotifycampaign.wordpress.com/tag/thatsongwhen/*

[4] *https://influencermarketinghub.com/influencer-marketing-benchmark-report-2020/*

[5] Kempe, D., Kleinberg, J., & Tardos, É. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 137-146, August, 2003.

[6] Lakhotia, K., & Kempe, D. Approximation algorithms for coordinating Ad campaigns on social networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 339-348, November, 2019.

[7] Nguyen, H., & Zheng, R. On budgeted influence maximization in social networks. IEEE *Journal on Selected Areas in Communications, 31*(6), 1084-1094, 2013.

[8] Giovanidis, A., Baynat, B., & Vendeville, A. Performance analysis of online social platforms. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pp. 2413-2421, April 2019.

[9] R. Srikant. *The Mathematics of Internet Congestion Control*. Springer Science & Business Media, 2004.

[10] Shakkottai, S., Shakkottai, S. G., & Srikant, R. *Network optimization and control.*. Now Publishers Inc., 2008

[11] Narici, L., & Beckenstein, E. *Topological vector spaces.*. CRC Press, 2010.

[12] *https://github.com/RLD-Hub/Social-Influencer-Selection*

[13] *https://www.kaggle.com/borisch/russian-election-2018-twitter*