



Contents lists available at ScienceDirect

## Expert Systems With Applications

journal homepage: [www.elsevier.com/locate/eswa](http://www.elsevier.com/locate/eswa)

## Heuristic optimization algorithms for advertising campaigns

Álvaro Seco<sup>a</sup>, Natalia López<sup>a</sup>, Fernando Rubio<sup>a,b</sup>\*<sup>a</sup> Dept. Sistemas Informáticos y Computación. Facultad de Informática Universidad Complutense de Madrid, 28040 Madrid, Spain<sup>b</sup> Instituto de Tecnología del Conocimiento. Universidad Complutense de Madrid, 28040 Madrid, Spain

## ARTICLE INFO

## Keywords:

Marketing  
 Advertisements  
 Optimization  
 Genetic algorithms  
 Heuristic methods  
 Approximability  
 NP-complete problems

## ABSTRACT

In this paper, two optimization problems within the scope of marketing campaign design are studied. In particular, two ad positioning problems are analyzed with the objective of minimizing the cost of all the chosen media while ensuring that a set of constraints is fulfilled. In the first problem, a given minimum number of impressions (views) in each population segment is required to be reached as a constraint. In the second problem, the constraints involve achieving, in each population segment, a given minimum probability that any individual will see the ad at least once. In this second case, media dependencies are defined for each population segment: independence, exclusion, and inclusion. Since both problems are Log-APX-hard, heuristic methods are used to solve them. More specifically, greedy and genetic algorithms, as well as particle swarm optimization, are applied to address these problems. Furthermore, the usefulness of these algorithms is evaluated through concrete case studies using real data on ad prices and views.

## 1. Introduction

During the last few years, computer science has been intensively used to deal with a multitude of problems within the broad area of social sciences (see e.g. Brandt, Conitzer, Endriss, Lang, and Procaccia (2016), Buchfuhrer, Schapira, and Singer (2010), Chiong et al. (2018), Das et al. (2016), Jin, Hu, Chan, et al. (2017), Keller, Hassidim, and Hazon (2019), Lewenberg, Lev, and Rosenschein (2017), Maymin (2011) and Chiong, Fan, Hu, and Dhakal (2022)), and it has also been used in the more particular area of marketing (see e.g. Balducci and Marinova (2018), Hurley, Moutinho, and Witt (1998), Lilien and Rangaswamy (2004), Liu and Ong (2008), Robles, Chica, and Cordón (2016), Souren, Ahn, and Schmitz (2005), Wedel and Kamakura (2000), Yunzhi, Yaoguang, Yanhua, and Ruijun (2010) and Robles, Chica, and Cordón (2020)). Due to the great complexity of many of these problems (for many of which even NP-hardness<sup>1</sup> has been proved, as shown in Rodríguez, Rubio, and Rabanal (2016) and Rodríguez, Rabanal, and Rubio (2017)), the use of heuristic computational methods is very useful to obtain practical solutions in real scenarios. In this paper, a complex problem is dealt with, where reasonable solutions are provided by computer science: the problem of media planning. In marketing, media planning is defined as the task of deciding where ads should be placed to effectively achieve campaign objectives. This task is carried out by selecting the most appropriate media for each occasion, and the lowest possible cost is always sought. Computational approaches

are provided to address two different optimization problems in media planning.

In the first problem, the costs of placing an advertisement in various media (for example, some television networks, some radio stations, some websites, etc.) are given, along with the number of impressions (views) of various segments of the population in each medium (for example, for men between 25 and 34, the impressions due to listening to a certain radio station are 25; for women between 55 and 64, the impressions for that same station are 58). Additionally, some target number of impressions to be reached for each of the population segments is specified (for example, for women between 35 and 44, at least 70 impressions must be reached across all media where the product is advertised; for men between 65 and 74, 45 impressions must be reached, etc.). The total cost of placing the ads is to be minimized while ensuring that the target impressions are reached in each of the established segments. It should be remarked that this problem reflects the most common situation encountered when an advertising campaign is carried out: a certain amount of the population is intended to be impacted, taking into account their profile. That is, depending on the specific product, it may be of interest for a greater impact to be achieved on, for example, young women, while another product may be especially oriented toward an older population.

The second problem is considered to be very similar to the first one. However, for each population segment, a given number of impressions

\* Corresponding author at: Dept. Sistemas Informáticos y Computación. Facultad de Informática Universidad Complutense de Madrid, 28040 Madrid, Spain.  
 E-mail addresses: [alseco@ucm.es](mailto:alseco@ucm.es) (Á. Seco), [natalia@sip.ucm.es](mailto:natalia@sip.ucm.es) (N. López), [fernando@sip.ucm.es](mailto:fernando@sip.ucm.es) (F. Rubio).

<sup>1</sup> A problem is NP-hard if any non-deterministic polynomial-time problem can be reduced in polynomial time to it.

<https://doi.org/10.1016/j.eswa.2024.126105>

Received 15 November 2022; Received in revised form 28 November 2024; Accepted 6 December 2024

Available online 13 December 2024

0957-4174/© 2024 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

is not required to be reached, but rather a probability that the ad will be watched at least once (for example, for women between 45 and 54, at least 30% of them must be reached; for men between 25 and 34, 10% must be reached, etc.). Due to probabilities being used, the best combination of media in which the ad is placed is largely determined by the correlation between the audiences of the different media. In this case, the most significant forms of audience dependencies (independence, exclusion, and inclusion) are differentiated. It should be noted that this second problem is considered to represent a tighter fit with respect to what a company would want to achieve. The reason is that in the first problem, a distinction cannot be made between an advertisement being seen 10 times by a single person and an advertisement being seen one time by 10 different persons. However, as will be discussed in the next section, this scenario is more difficult to handle.

In order to clarify the differences between both problems, a simple example can be considered. A product is assumed to be advertised among a population of twelve million inhabitants. Three options are provided for including the ads: (i) a soccer match watched by one million inhabitants; (ii) the post-match special program, where the best moments are relived, watched by four hundred thousand people; and (iii) a documentary on wild nature, broadcast at the same time as the soccer match and watched by only two hundred thousand people. If the goal is for the ad to be seen 1,300,000 times, advertising during the match and post-match must be selected. This scenario is reflected in the first problem described above, where a certain number of impressions is required without concern for whether the same viewer sees the ad multiple times. Now, suppose a probability of 10% is desired for the ad to be seen by each individual, meaning 10% of the population must watch the ad at least once. In this case, it would be expected that the vast majority of the post-match viewers would have also watched the match. Therefore, although 1,400,000 impressions would be achieved between the two events, the number of distinct viewers would be expected to be around one million, which is far from 10% of the population. If the ad is placed during the match and the documentary instead, 1,200,000 impressions would be achieved, and 1,200,000 distinct viewers (10% of the population) would be guaranteed. This is because no one could watch both the soccer match and the documentary at the same time. In other words, more people would be reached, even though the total number of views would be lower than if the ad were placed in the match and post-match. This is the type of scenario pursued in the second problem, for which information about the dependency relationships between the different media where ads can be placed is required. In this example, an inclusion relationship is observed between watching the post-match and watching the match, while an exclusion relationship is noted between watching the match and watching the documentary.

It is known that both problems are not only NP-hard, but also difficult to be approximated. In particular, it has been proved that the problems are Log-APX-hard<sup>2</sup> (see Rodríguez et al. (2016)). Consequently, a polynomial-time algorithm that guarantees a given quality ratio for any input problem cannot be implemented (unless P=NP). However, genetic algorithms (GA) (De Jong, Spears, et al., 1989; Katoch, Chauhan, & Kumar, 2021; Mitchell, 1998) are considered good candidates for providing reasonable solutions in these cases. More precisely, as shown in Muñoz and Rubio (2021), a combination of a greedy ad-hoc heuristic and a genetic algorithm can be the best option for dealing with problems belonging to Log-APX approximation class. Thus, for each problem, a greedy heuristic is used, which is defined as a deterministic algorithm that aims to find an optimal solution to a problem by making locally optimal choices (Curtis, 2003), along with a GA.

In order to evaluate the usefulness of the implementations, two concrete case studies are dealt with. The design of the case studies has been

<sup>2</sup> Log-APX is the class of problems that can be approximated (using polynomial-time algorithms) within a factor that grows logarithmically with respect to the input size.

carried out using real data. In particular, the information on the impact of each media on each segment of the population has been extracted from the Spanish EGM,<sup>3</sup> a multimedia study responsible for analyzing media consumption in Spain and the behavior of the population with respect to different media. EGM publishes data covering magazines, television, radio and web pages. This sample comprises information from more than 150,000 individuals, based on telephone, online and face-to-face interviews. Moreover, the price of including an advertisement in each possible media has been established based on official data on advertising rates in each media. Taking all this information into account, two independent advertising campaigns are proposed, aimed at different types of customers and with different impact objectives. The results obtained in both cases illustrate that genetic algorithms can be used to achieve good results within a reasonable time frame.

The main contributions of the paper are the following:

- To provide a formal definition of two relevant problems in the area of economic optimization of advertising campaigns.
- To provide heuristic algorithms that solve such problems.
- Application of these heuristic algorithms to solve case studies where real-world data are being used.

The rest of the paper is structured as follows. In the next section, the marketing problems are formally defined. Then, in Section 3 related works are summarized. Next, in Section 4, the algorithms used to solve them are presented. Afterwards, in Section 5 the data used to define the case studies analyzed in the experiments of Section 6 are introduced. Finally, in Section 7, the conclusions are presented, along with lines for future work.

## 2. Problem formulation

In this section, the problems are formally introduced. In the first case (Advertisement Placement Optimization, APO), a given number of impressions is aimed to be obtained in each population segment. In the second case (Probability Advertisement Placement Optimization, PAPO), it is desired that each individual reaches a certain probability of seeing the ad. After presenting such problems, two concrete marketing campaigns are introduced: a video game company and a hearing aid store.

### 2.1. APO

The first problem is to determine where the ads should be placed, with the objective of minimizing the total cost, while ensuring that at least a certain number of impressions established for each population segment is reached. It should be noted that a vector  $b_1 \dots b_m$  denoting the minimum number of times the advertisement should be viewed by each population segment  $S_1 \dots S_m$ , is provided as input. Additionally,  $n$  different media  $M_1 \dots M_n$ , where advertisements can be placed, are also given as input. Furthermore, for each media  $M_i$ , the cost of placing an advertisement ( $c_i$ ) is known, along with the number of people who will view such an advertisement. Specifically, for each media  $M_i$  the number of people from each population segment  $S_j$  who will view the advertisement is also known. For example,  $a_{ij}$  represents that media  $M_i$  is watched by  $a_{ij}$  individuals belonging to the population segment  $S_j$ . Taking into account all this information, the problem can be formally defined as follows:

**Definition 2.1.1 (APO).** Let  $(M_1, \dots, M_n)$  be  $n$  media where the product can be advertised, and  $(S_1, \dots, S_m)$  be  $m$  population segments. Let  $c \in \mathbb{N}^n$  be a vector of naturals where each value  $c_i \in \mathbb{N}$  denotes the cost of advertising the product in media  $M_i$ ,  $A \in \mathbb{N}^{m \times n}$  be a matrix where

<sup>3</sup> EGM stands for *Estudio General de Medios*, whose translation is General Media Study.

each value  $a_{ij} \in \mathbb{N}$  denotes the number of impressions by customers of segment  $S_i$  due to advertising the product in media  $M_j$ , and  $b \in \mathbb{N}^m$  be a vector where each value  $b_j \in \mathbb{N}$  denotes the minimum number of impressions required to be reached for customers in segment  $S_j$ .

The **Advertisement Placement Optimization** problem, denoted by **APO**, is the problem of minimizing  $c^T \cdot x$  subject to  $x \in \{0, 1\}^n$  and  $A \cdot x \geq b$ .

## 2.2. PAPO

Next, the second optimization problem is defined. In the previous problem (APO), a certain minimum number of impressions for each population segment was aimed to be achieved by summing the impressions of all individuals in the segment. This can be achieved by having many people seeing the ad only once, or by having a few people seeing the ad many times. Therefore, the solutions obtained in APO do not necessarily show that the ads are seen by a large number of people. However, in this second problem, the focus is placed on ensuring that, in each population segment, a minimum probability with which an individual sees the ad at least once is obtained. Thus, one individual who sees the ad multiple times does not compensate for others who do not see it, as was the case in APO.

Using the probabilities as objectives to be met, instead of a number of expected impressions, requires the introduction of new factors into the problem. In order to calculate the probability of the ad being watched at least once when it is advertised in  $M_1$  and  $M_2$ , the degree of dependency between the events “watch  $M_1$ ” and “watch  $M_2$ ” must be considered. Let  $e_i$  be the event “watching  $M_i$ ”. If  $n$  events  $e_1, \dots, e_n$  are considered, then, by the Inclusion–Exclusion Principle, the following expression is obtained:

$$P\left(\bigcup_{1 \leq i \leq n} e_i\right) = \sum_{1 \leq k \leq n} (-1)^{k-1} \left( \sum_{I \subset \{1, \dots, n\}, |I|=k} P\left(\bigcap_{j \in I} e_j\right) \right)$$

Therefore, the calculation of the probability of the ad being seen at least once, if  $M_1, \dots, M_n$  are selected, requires that the probability of simultaneously seeing any pair of media, any triplet of media, and any other tuple of media up to  $n$ , be known. All these combinations are important to determine  $P(\bigcup_{1 \leq i \leq k} e_i)$  with  $k \in \{1, \dots, n\}$ . Unfortunately, media and audience measurement techniques cannot reliably provide the necessary information on all combinations. As an alternative, a set of specific event dependency cases will be used. These cases are not fully representative but express sufficiently well the most relevant dependencies:

**(a) Independence** of events  $e_1, \dots, e_n$ , denoted by  $\mathbb{D}(e_1, \dots, e_n)$ , i.e., it is assumed that the probability of any event  $e_i$  is not dependent on whether any other event  $e_j$  with  $i \neq j$  occurs or not. In this particular case, the following expression is obtained:

$$P\left(\bigcup_{1 \leq i \leq n} e_i\right) = 1 - ((1 - P(e_1)) \cdot \dots \cdot (1 - P(e_n)))$$

**(b) Exclusion** of events  $e_1, \dots, e_n$ , denoted by  $\mathbb{O}(e_1, \dots, e_n)$ , i.e., it is assumed that any pair of events  $e_i$  and  $e_j$  with  $i \neq j$  cannot occur together. In this case, the following expression is obtained:

$$P\left(\bigcup_{1 \leq i \leq n} e_i\right) = P(e_1) + \dots + P(e_n)$$

**(c) Inclusion** of event  $e_i$  into event  $e_j$ , denoted by  $\mathbb{C}(e_i, e_j)$ , i.e., it is assumed that  $e_i$  is not possible without  $e_j$ . In this case, the following expression is obtained:

$$P(e_i \cup e_j) = P(e_j)$$

To calculate the probability that an individual in a population segment sees the ad at least once, the dependencies between viewing each of the media in which the ad is included need to be known. Let  $x = (x_1, \dots, x_n)$  be a tuple of bits where  $x_i \in \{0, 1\}$  denotes whether or not the advertisement is placed in  $M_i$ . Given the probability that a person sees  $M_i$ , denoted by  $P(e_i)$ , the media where the ads are placed, i.e.,  $x$ , and the dependency constraints between media, the probability that a person sees at least one of the media in which the ad has been placed will be computed. This probability will be calculated recursively, following the structure of the dependency constraints. The dependency constraints will be represented as trees in which all events appear only once.

**Definition 2.2.1 (Probability of Watching the Advertisement).** For all  $1 \leq i \leq n$ , let  $e_i$  be the event of watching media  $M_i$ ,  $P(e_i) \in [0, 1]$  be its probability and  $x_i \in \{0, 1\}$  denote whether the advertisement is placed in  $M_i$  or not. Let  $t$  be a well-formed dependency constraint for events  $e_1, \dots, e_n$ . Let  $P = (P(e_1), \dots, P(e_n))$  and  $x = (x_1, \dots, x_n)$ . The *probability of watching the advertisement* under  $t$ ,  $P$  and  $x$ , denoted by  $wAd(t, P, x)$ , is defined as:

$$wAd(t, P, x) = \begin{cases} 1 - \prod_{1 \leq k \leq m} (1 - wAd(t_k, P, x)) & \text{if } t = \mathbb{D}(t_1, \dots, t_m) \\ \sum_{1 \leq k \leq m} wAd(t_k, P, x) & \text{if } t = \mathbb{O}(t_1, \dots, t_m) \\ P(e_i) & \text{if } t = \mathbb{C}(t', e_i) \wedge x_i = 1 \\ wAd(t', P, x) & \text{if } t = \mathbb{C}(t', e_i) \wedge x_i = 0 \\ P(e_i) & \text{if } t = e_i \wedge x_i = 1 \\ 0 & \text{if } t = e_i \wedge x_i = 0 \end{cases}$$

All the data and operations required to define the second problem are now available. As in the case of APO, input parameters will include information about the cost ( $c_j$ ) of placing advertisements in each media  $M_j$ . The difference is that the objective is no longer the number of impressions in each population segment, but rather a probability  $mP_i$  of being watched in each population segment  $S_i$ . Therefore, input must include the probability  $p_{ij}$  of being watched by each population segment  $S_i$  in each media  $M_j$ . Additionally, as explained earlier, the dependencies  $t_i$  among different media for each population segment  $S_i$  must also be known. Taking all this information into account, the problem can be formally defined as follows:

**Definition 2.2.2 (PAPO).** Let  $(M_1, \dots, M_n)$  be  $n$  media where the product can be advertised and  $(S_1, \dots, S_m)$  be  $m$  population segments. Let  $c \in \mathbb{N}^n$  be a vector of naturals where each value  $c_i \in \mathbb{N}$  denotes the cost to advertise the product in media  $M_i$ . For all  $1 \leq i \leq m$  let  $P_i = (P_i(e_1), \dots, P_i(e_n))$  denote the probabilities that individuals in segment  $S_i$  watch each media  $M_1, \dots, M_n$ , respectively, so  $P \in [0, 1]^{m \cdot n}$ . Let  $t_i$  denotes the dependency constraint for individuals in segment  $S_i$ , and  $mP_i$  be the target probability for the advertisement to be watched by individuals in  $S_i$  at least once.

The **Probability Advertisement Placement Optimization** problem, denoted by **PAPO**, is the problem of minimizing  $c^T \cdot x$  subject to  $x \in \{0, 1\}^n$  and  $wAd(t_i, P_i, x) \geq mP_i$  for all  $1 \leq i \leq m$ .

## 2.3. Marketing campaigns to be considered

As an illustration of the problems described in the previous sections, two independent advertising campaigns are considered. One involves the application of the APO problem, while the other requires addressing the PAPO problem. For each campaign, a product is advertised while adhering to a series of budget and viewership restrictions.

In the first experiment, the role of a video game company is assumed. Specifically, the launch of a new video game aimed at a young audience is simulated, with the objective of achieving a high number of impressions among the youngest age groups. For each population

segment, the number of impressions to be reached is established by setting a percentage of the total number of possible impressions for that segment (*Impressions%*). It should be noted that this does not mean that the ad will be seen by that percentage of the segment. Instead, if 100,000 impressions can be achieved by including the ad in all possible media, and an *Impressions%* target of 15% is set, the ad should be seen 15,000 times, though not necessarily by 15,000 different individuals.

In this experiment, the *Impressions%* to be reached decreases progressively as the age increases, until no impressions are targeted for individuals over 65. Furthermore, according to AEVI<sup>4</sup> in 2020 46% of video game consumers were women. Therefore, the same targets are used for both males and females in this test.

In the second experiment, the promotion of a hearing aid store is simulated. It is assumed that the store has an established reputation, and the objective is to ensure that any individual has a given probability of being exposed to the advertisement at least once.

According to the SEORL-CCC,<sup>5</sup> hearing loss caused by age affects 20% of the population over 60 years, 30% between 65 and 70 years, and 70% of the people over 70 years. Therefore, the campaign is focused on covering these age groups, without considering differences between males and females. In addition, the offspring of seniors are also targeted, though with a lower probability, as they may want to purchase hearing aids for their parents. Thus, adult age groups are also included in the campaign.

In both experiments, three different scenarios will be conducted, representing campaigns aiming to achieve lower or higher objectives. These scenarios range from a large company with a substantial budget to a smaller company capable of affording only a modest advertising campaign.

### 3. Related works

Optimization in the design of marketing campaigns is a topic of great current interest, being addressed from several perspectives (see Yun et al. (2020)). All of them are partially related to the work presented in this paper. To begin with, a multitude of studies aim to improve customer segmentation and predictive analysis techniques. For example, in Chan, Hwang, and Wu (2016), a method based on Particle Swarm Optimization is developed that manages to obtain customers clusters with higher degrees of affinity compared to those produced by more classical techniques. In general, machine learning techniques are proving highly useful for classifying and analyzing users, both when providing general approaches (see e.g. Hicham and Karim (2022)) and when targeting more specific market niches, such as video games, as discussed in Waqas (2020). As highlighted in Arora and Thota (2024), these techniques have facilitated the development of personalized marketing strategies.

Optimization in online advertising campaigns constitutes a particularly active sub-area (see e.g. Nuara (2020)). Within this context, works such as Barajas et al. (2012) analyze methods for evaluating the success of a strategy. Others, such as Nuara (2020), focus more on designing strategies for auctions where real-time bid optimization is required while maintaining the pre-established budgets. The use of genetic algorithms is particularly interesting for the optimization of all kinds of problems. For example, in Miralles-Pechuán, Ponce, and Martínez-Villaseñor (2018) a micro-targeting technique is proposed to optimize the click-through rate (CTR) when displaying ads in online media.

Even when the focus is on marketing campaigns in more traditional media (radio, television, newspapers, magazines, etc.), algorithmic marketing is overtaking traditional marketing techniques (see

e.g. Degterev (2019)). In this context, works such as Han (2020) present an interesting comparative analysis examining the evolution from traditional marketing to big-data-driven marketing. This study highlights the necessity for companies to adapt to real-time marketing strategies and develop new data collection methods.

Another closely related aspect of this work is the more theoretical analysis of the computational complexity of different problems in the area of advertising campaign design. For example, in Chen, Li, Lin, and Rubinstein (2015) the NP-hardness of a problem that combines traditional marketing techniques together with viral marketing techniques is proven. Additionally, it is demonstrated that polynomial-time algorithms cannot guarantee good approximation ratios unless  $P=NP$ . A similar example is found in Zehmakan, Zhou, and Zhang (2023), which establishes the inapproximability of viral marketing predictability in social networks where there are several competing products. As shown in Serrano and Blum (2022), such inapproximability proofs justify the need of heuristic methods such as genetic algorithms or swarm intelligence, as we propose in the present work.

### 4. Algorithms

In this section, three algorithms are presented to address APO, along with three variations to address PAPO. The first algorithm is based on a greedy method, the second is implemented using a genetic algorithm, and the third is designed using particle swarm optimization.

#### 4.1. Greedy algorithms

To solve APO (and also PAPO), the greedy method introduced in Chvátal (1979) for addressing the minimum set cover problem (MSC) when each subset has a given weight is adapted. This greedy algorithm is considered the best approximation algorithm that can be achieved in polynomial time for the most general case of solving MSC (see e.g. Dinur and Steurer (2014), Vazirani (2001) and Solomon and Uzrad (2023)). It should be noted that MSC is one of the most significant representatives of the Log-APX-complete approximability class. Any other Log-APX-complete problem can be polynomially reduced to MSC while preserving approximability. For this reason, the greedy algorithm that approximates MSC has been adapted to create new greedy algorithms for approximating the APO and PAPO problems. The algorithm proceeds by iterating the following steps until all the constraints are satisfied:

1. The variable  $x_k$  is selected based on the highest ratio between its cost and its contribution to satisfying all the inequalities implicitly imposed by  $A \cdot x \geq b$  (APO) or  $wAd(t_i, P_i, x) \geq mP_i$  (PAPO). This contribution is measured in terms of the amount added to the left-hand side of all inequalities when  $x_k = 1$  is set.
2. The variables  $yetToBeCovered_i$  are defined to account how much remains to satisfy each inequality.
3. If for all  $1 \leq i \leq m$ ,  $yetToBeCovered_i = 0$ , the impression requirements (APO) or probability requirements (PAPO) are full-filled, and the algorithm is completed.

It should be noted that the same structure is exactly used for both APO and PAPO, but using different formulas to compute the distance to the required constraints.

For the sake of simplicity, the case of APO is considered. First, for each variable  $x_k$ , its contribution to each population segment is computed. The number of impressions obtained in each population segment when  $x_k$  is set to 1 (i.e. when an advertisement is paid for in media  $k$ ) is known, as well as the number of impressions still required in each population segment. For instance, suppose that 1000 impressions are still needed in segment  $S_1$ . If  $x_3$  provides 300 impressions in segment  $S_1$ , then its contribution is 300, and 700 more impressions will still be needed. However, if  $x_4$  provides 5000 impressions in segment  $S_1$ , then its contribution is 1000 (not 5000) because only 1000 impressions are

<sup>4</sup> Asociación Española de Videojuegos, Spanish Association of Video Games.

<sup>5</sup> Sociedad Española de Otorrinolaringología y Cirugía de Cabeza y Cuello, Spanish Society of Otorhinolaryngology and Head and Neck Surgery.

needed in total in segment  $S_1$ . Once the contribution in each segment is determined, all contributions are aggregated to compute the overall contribution to all segments. This value is then divided by the cost of advertising in media  $k$ , resulting in the contribution per unit of money. In each step,  $x_k$  is selected so that its overall contribution per unit of money is the highest.

#### 4.2. Genetic algorithms

In addition to the previous described greedy algorithms, genetic algorithms are also used to solve both problems. A binary representation is used for the genes, where each gene represents whether the corresponding media is used to advertise the product or not. In a chromosome the  $i$ th position indicates whether or not it is advertised in the  $i$ th media. For example, if five possible media are considered (Vogue, Telva, Cosmopolitan, National Geographic, Men's Health), then chromosome 01101 would represent that advertisements are placed in Telva, Cosmopolitan and Men's Health. Similarly, the chromosome 10100 represents that advertisements are placed in Vogue and Cosmopolitan. The pseudocode for the genetic algorithm employed is presented in the following code snippet:

```
-----
population = create_initial_random_population()
# Add the greedy solution to the population (optional)
population = population + greedy_solution
population_ordered = order_by_fitness(population)
for generation in range(iterations):
    best_solution_so_far = population_ordered[0]
    selected_individuals = selection(population_ordered)
    descendants = crossover(selected_individuals)
    new_population = replace(population_ordered, descendants)
    new_population_mutated = mutation(new_population)
    population_ordered = order_by_fitness(new_population_mutated)
    if elitism
        applyElitism(population_ordered, best_solution_so_far)
best_solution = population_ordered[0]
-----
```

Listing 1: Pseudocode of the Genetic Algorithm

As in any genetic algorithm, the fitness function to be minimized must be defined. To evaluate each individual, the variables  $yetToBeCovered_i$  ( $1 \leq i \leq m$ ), mentioned in the greedy algorithms, are taken into account. That is, if the chromosome satisfies the condition  $yetToBeCovered_i = 0$ , for all  $1 \leq i \leq m$ , it indicates that all constraints are satisfied. In this case, the fitness function is defined as the sum of the costs of the media that are active in the chromosome. Otherwise, if the constraints are not satisfied, and the fitness function returns  $\sum_{j=1}^n c_j + \sum_{i=1}^m yetToBeCovered_i$ , which represents the total cost of all media plus what it is still left to be covered. By doing so, the value returned will be greater than the cost of any individual that satisfies the constraints. Furthermore, the farther an individual is from satisfying the constraints, the higher its value will be. So, the *fitness* function is defined as:

$$fitness(x_1, \dots, x_n) = \begin{cases} \sum_{j=1}^n c_j \cdot x_j & \text{if } \forall 1 \leq i \leq m, yetToBeCovered_i = 0 \\ \sum_{j=1}^n c_j + \sum_{i=1}^m yetToBeCovered_i & \text{otherwise} \end{cases}$$

The main configurations used for the experiments are described next, although more details about the selection of such configurations will be provided in Section 6. Following the method commonly employed in the field (see e.g. De Jong (2007), Qasem and Lam (2023) and Cura (2023)) experiments have been performed with different configurations, and non-parametric statistical tests have been used to analyze the differences between them (see e.g. Chiarandini, Basso, and Stütze (2005) and Derrac, García, Molina, and Herrera (2011)).

In general, very small or very large population sizes can lead to poor experimental results (see e.g. Chen, Tang, Chen, and Yao (2012)). It is usually recommended that sizes be proportional to the number

of genes (see e.g. Lobo and Lima (2005)). However, to determine an optimal population size, it is customary in any population-based search method to test a set of possible values and analyze which ones are best suited to the problem, performing a statistical study with a sufficiently large sample of runs (see e.g. Olympia, Stefka, and Paprzycki (2013) and Piotrowski, Napiorkowski, and Piotrowska (2020)). In particular, different population sizes, ranging from 80 to 250 individuals, have been tested, with the best results obtained using 150. Regarding the initialization of the population, all individuals except one are randomly generated. However, following the recommendations in Muñoz and Rubio (2021), the initial population also includes an individual corresponding to the solution obtained by the greedy algorithm.

Once the initial population is obtained, the process of creating new generations begins. Experiments have been conducted using different selection strategies, including roulette wheel selection, tournament selection, and truncation selection. For each strategy, both the introduction of elitism and its absence have been considered. By adding elitist selection to this method, it is ensured that the best chromosome from the previous generation is retained if the population does not improve with a new iteration, allowing it another opportunity to reproduce and mutate. As shown in Section 6, the use of elitism improves the obtained results. Furthermore, both roulette wheel and tournament selection achieve similar results, clearly outperforming those obtained using truncation selection.

After the chromosomes are selected, reproduction is carried out. In this case, three alternative strategies have been tested: one-point crossover, two-points crossover, and uniform crossover, where each gene of the descendant is randomly acquired from one of the two parents. Experiments indicate that all of these strategies yield similar results.

Finally, different mutation probabilities ranging from 1% to 30% have been tested, with the best results obtained using a mutation probability of 15%. In case a chromosome is to be mutated, at least one of its genes will be modified (using a binary representation, its current value is simply negated). Additionally, several experiments have been conducted allowing changes from 1 up to 10 genes of each chromosome, with the best upper limit found to be 3.

#### 4.3. Particle swarm optimization

In addition to the previous methods, an implementation based on Particle Swarm Optimization (PSO) has been provided, a method inspired by the movement observed in flocks of birds or shoals of fish, as described in Kennedy and Eberhart (1995) and Gad (2022). Specifically, a binary version of PSO, called BPSO, is used. In this version, the individuals (possible solutions) in the population are represented in the same way as in the genetic algorithms. That is, a possible solution is a vector of bits, where the  $i$ th bit is set to 1 if and only if the  $i$ th communication medium is selected.

The same fitness function as in the genetic algorithms is used, and all the auxiliary functions  $yetToBeCovered_i$  can be reused. The main difference lies in the strategy employed by BPSO to explore the solution space. In genetic algorithms the population evolves through recombination of previous individuals. In contrast, in PSO, each individual in the population has a velocity/direction vector that indicates how its value will change between iterations. This direction vector is updated at each step, taking into account a certain inertia, as well as a tendency to move toward the best solution found so far by the individual itself, and a tendency to move toward the best solution found so far by the entire swarm.

The fundamental PSO algorithm was originally designed for continuous optimization problems, where particles move through an  $n$ -dimensional real-valued space. In the continuous version, each particle  $i$  maintains three key vectors: its current position  $\vec{x}_i$ , its velocity  $\vec{v}_i$ , and its personal best position found so far  $\vec{p}_i$ . Additionally, the algorithm

keeps track of the global best position  $\vec{g}$  found by any particle in the swarm. The movement of particles is governed by two main equations:

$$\vec{v}_i(t+1) = w\vec{v}_i(t) + c_1r_1(\vec{p}_i - \vec{x}_i(t)) + c_2r_2(\vec{g} - \vec{x}_i(t)) \quad (1)$$

$$\vec{x}_i(t+1) = \vec{x}_i(t) + \vec{v}_i(t+1) \quad (2)$$

where  $w$  is the inertia weight,  $c_1$  and  $c_2$  are the cognitive and social parameters respectively, and  $r_1, r_2$  are random numbers uniformly distributed in  $[0, 1]$ . The first term represents the particle's momentum, the second term pulls the particle toward its personal best position, and the third term pulls it toward the global best position.

For discrete binary optimization problems, like the one addressed in this work, the Binary PSO (BPSO) adaptation is required. In BPSO, while the velocity update remains similar to the continuous version, the position update mechanism is modified to handle binary values. The main difference lies in how the velocity is used to determine position changes. Instead of directly adding the velocity to the position, BPSO uses a sigmoid function to transform the velocity into a probability of bit flipping:

$$S(v_{ij}) = \frac{1}{1 + e^{-v_{ij}}} \quad (3)$$

The position update rule then becomes:

$$x_{ij}(t+1) = \begin{cases} 1 & \text{if } rand() < S(v_{ij}(t+1)) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where  $x_{ij}$  represents the  $j$ th bit of the  $i$ th particle's position,  $v_{ij}$  is the corresponding velocity component, and  $rand()$  generates a random number in  $[0, 1]$ . This transformation ensures that higher absolute velocities result in a higher probability of the bit being flipped to 1, while velocities near zero maintain the current bit value with higher probability.

The pseudocode for the BPSO employed is presented in the following code snippet:

```

-----
# Initialize the population
swarm = initialize_random_particles()
if greedy:
    swarm.add(greedy_solution)
for iteration in range(iterations):
    for particle in swarm:
        # Update particle fitness
        particle.fitness = evaluate_fitness(particle.position)
        # Update personal best position and fitness
        if particle.fitness < particle.personal_best_fitness:
            particle.personal_best_position = particle.position
            particle.personal_best_fitness = particle.fitness
        # Update global best position and fitness
        global_best = get_global_best(swarm)
        # Update particle velocities and positions
        for particle in swarm:
            particle.velocity = update_velocity(particle, global_best,
                                                params)
            particle.position = update_position(particle, bounds)
    best_solution = global_best.position
-----

```

Listing 2: Pseudocode of the BPSO Algorithm

The procedure shown in the previous pseudocode implements these velocity and position update rules, where `update_velocity` applies Equation (1) and `update_position` implements the sigmoid transformation and probabilistic bit-flipping mechanism described in Eqs. (3) and (4).

As in the case of genetic algorithms, the solution of the greedy algorithm can be included as one of the original individuals in the swarm, or all individuals can be generated randomly. Experimental results show that including the greedy solution to guide the BPSO search yields better outcomes. In fact, a statistically significant difference is observed between using the greedy solution and not using it.

Regarding the specific parameters used for BPSO, the recommendations provided in Pedersen (2010) have been followed. These recommendations were determined through meta-optimization across multiple benchmark functions and selected based on their proven effectiveness across various optimization scenarios, particularly for problems with moderate dimensionality and sufficient fitness evaluations. In particular, the tests indicate that an effective parameter configuration includes setting the inertia to 0.5, which helps balance exploration and exploitation. The social component is set to 1.1, determining the particle's tendency to move toward the global best position, and the cognitive component is set to 1.6, influencing the particle's inclination to move toward its personal best position. Notably, the cognitive component is set higher than the social component ( $1.6 > 1.1$ ) to favor self-positioning and exploration, thus avoiding quick convergence into local maxima. Additionally, the configuration includes a population size of 150 individuals, ensuring sufficient diversity in the search space while maintaining computational efficiency. Furthermore, 30 neighbors are considered to enable adequate information sharing among particles without leading to premature convergence.

## 5. Definition of case studies

In this section, the real media framework used to set up the experiments is presented. After brief description of the population segments considered, the media are shown, including information about both the cost of advertising in them and the impacts that would be obtained in each population segment. In addition, the existing viewing dependencies between the different media and for each population segment are presented.

### 5.1. Population segments

In the experiments,  $m = 16$  population segments are considered. These segments are differentiated by age (divided into eight age intervals) and gender, as follows:

Age	14–19	20–24	25–34	35–44	45–54	55–64	65–74	75+
Gender	♂ ♀	♂ ♀	♂ ♀	♂ ♀	♂ ♀	♂ ♀	♂ ♀	♂ ♀

The 16 population segments are considered because the actual data available, obtained from the Spanish General Media Study, is classified using exactly these same segments, plus an additional child segment. It should be noted that only individuals aged fourteen years or older are considered, as advertising campaigns targeting children are not conducted. Thus, the segments used represent the most accurate population data accessible for advertising campaigns.

### 5.2. Media

A total of  $n = 65$  different media are considered for publishing advertisements. These media represent some of the most well-known media in Spain and are classified into six categories: magazines, newspapers, radio stations, national television channels, local television channels, and websites. For each of them, not only is the price of publishing an ad known, but information on the number of expected impacts in each population segment is also available. This information has been obtained through the Tom Micro software, developed by a company called ODEC. Tom Micro is a software that provides tools to aid media selection planning for advertising campaigns (ODEC, 2022). This software provides the information gathered from the EGM, a multimedia study responsible for analyzing both media consumption in Spain and the behavior of the population across the different media (AIMC, 2022). Specifically, EGM data from 2021, collected between

September and December, is used. From the data provided by the tool, information about GRP's (Gross Rating Points Fulgoni, 2015) and Coverage% is utilized. Using this data, the total impressions obtained in each segment are calculated as follows:

$$Impressions(segment) = \frac{Total\ Number\ Of\ People(segment)}{100} * GRP's$$

Regarding the probabilities, the Coverage% provided by the software is directly used. Coverage% is defined as follows (DBi Data Business Intelligence, 2020):

$$Coverage\%(segment) = \frac{People\ Who\ Watched\ The\ Ad\ At\ Least\ Once(segment)}{Total\ Number\ Of\ People(segment)}$$

Finally, the real price of advertising in each media is obtained through this tool. It is worth noting that the costs have been slightly simplified, retaining only the typical costs for each media. In particular, for newspapers and magazines, the possibility of placing ads on the front and back covers (which have significantly higher costs than inside pages) has been excluded. For television channels, the price of advertising varies considerably depending on the time slot, so each time slot is considered as if it were a separate media.

The selected media and the cost of placing an advertisement in each of them are shown below.

Magazines	Cost per ad (€)	Newspapers	Cost per ad (€)
¡Hola!	20.125	20 Minutos	37.875
Semana	10.745	ABC	23.811
Diez Minutos	11.725	As	15.350
Pronto	18.430	El Economista	7.200
Sport Life	5.775	El Mundo	37.200
Men's Health	10.500	El País	46.775
National Geographic	19.050	La Razón	16.680
Cosmopolitan	16.625	La Vanguardia	27.500
Vogue	19.900	Marca	28.100
Telva	19.850	Mundo Deportivo	8.225

Radio	Cost per ad (€)	Website	Cost per ad (€)
Cadena SER	8.500	elpais.com (El País)	37.000
Los 40 Principales	4.616	lavanguardia.com (La Vanguardia)	25.000
Los 40 Classic	2.292	as.com (As)	37.000
Cadena DIAL	3.711	mundodeportivo.com (El Mundo Deportivo)	25.000
Cadena 100	4.086	ondacero.es (Ondacero)	19.980
Ondacero	6.823	antena3.com (Antena 3)	57.400
COPE	7.645	eleconomista.es (El Economista)	25.000
Radio Marca	1.082	cincodias.es (Cinco Días)	18.000
Europa FM	3.251	europafm.com (Europa FM)	11.100
Kiss FM	3.091	lasexta.com (La Sexta)	22.960

National TV	Period	Cost per ad (€)	Period	Hours
Antena 3	Morning	1.883	Morning	8:00–14:00
	Noon	10.150	Noon	14:00–16:00
	Afternoon	8.000	Afternoon	16:00–20:00
	Night	19.143	Night	20:00–24:00
La Sexta	Morning	1.290	Local TV	Cost per ad (€)
	Noon	3.500		
	Afternoon	3.100		
	Night	11.000		
Neox	Morning	1.250	TV3	4.183
	Noon	2.100	Canal SUR	3.091
	Afternoon	2.000	Aragón TV	454
Nova	Night	3.600	TV de Galicia	1.226
	Morning	192	TV Canaria	502
	Noon	1.100		
	Afternoon	1.625		
Disney Channel	Night	2.814		
	Morning	560		
	Noon	1.050		
Channel	Afternoon	1.195		
	Night	1.870		

### 5.3. Dependency constraints among media

To address PAPO instances, the dependencies among different media need to be defined. These dependencies, however, can vary for each population segment. In particular, the population segments are grouped into three major categories, each of which is assigned its own dependencies:

Group	Youth/Students	Adults/Workers	Seniors/Retired
Age	14–19	20–24 25–34 35–44 45–54 55–64	65–74 75+

Although in general the dependencies are almost the same in all groups, subtle differences are introduced due to factors such as time availability resulting from work or media access habits influenced by age. These differences mean that the probabilities of any individual watching the ad at least once differ across segment. Thus, they provide variety and reliability within each segment.

As an example, Fig. 1 summarizes the dependencies among newspapers.<sup>6</sup> Newspapers are divided into four subcategories (daily news, sports, economy, and free newspapers), and it is assumed that these categories are independent within any population segment. That is, reading a sports newspaper is independent of reading an economy newspaper, as the content of both is unrelated. Furthermore, exclusivity is assumed within each subcategory. That is, individuals are expected to read only one sports newspaper, not multiple, as all of them essentially contain the same information. However, differences among population segments arise when considering newspapers with a digital counterpart. In these cases, Youth/Students (who are much more familiar with new technologies) are assumed to exhibit an inclusion between newspapers and their digital form. That is, it is assumed that only those who visit a newspaper's website would also be interested in the printed

<sup>6</sup> Similar dependencies have been defined for each type of media. However, for the sake of clarity, all details are not shown here. Interested readers can find them in <https://github.com/alvaroseco/Using-genetic-algorithms-for-advertising-campaigns>.

Youth/Students & Adults/Workers

Seniors/Retired

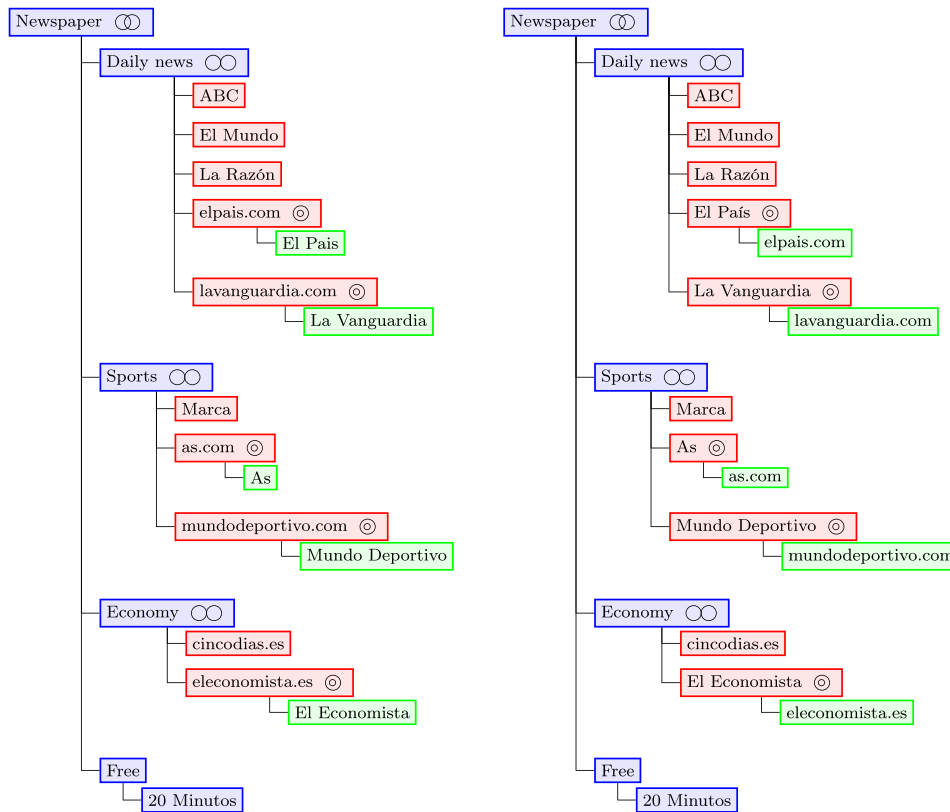


Fig. 1. Example of dependency constraints for newspapers in different population segments.

version of the newspaper. For Adults/Workers, same dependencies as for the Youth/Students are applied. However, unlike young people and adults, older individuals are less familiar with new technologies. For this reason, a different inclusion relationship is assumed between newspapers with a digital counterpart: it is presumed that only those who read the printed version of the newspaper would also be interested in its corresponding website.

6. Experimental results

In this section, the experimental results obtained for two specific advertising campaigns are presented. In both cases, the population segments and media described in the previous section are used. For the first campaign, the focus is placed on the APO problem, while for the second, the objectives are defined in terms of the PAPO problem. The algorithms have been implemented in the Python programming language and can be freely downloaded from <https://github.com/alvaroseco/Using-genetic-algorithms-for-advertising-campaigns>.

The experiments will illustrate the functionality, efficiency and applicability of the algorithms.

6.1. Statistical considerations

Unfortunately, theoretical studies are unable to establish the optimal configurations of genetic algorithms when solving complex optimization problems. Some interesting theoretical works are limited to study very specific and relatively simple cases. For example, in Chen et al. (2012), an extreme case is studied where only one global optimum and one local optimum are located in opposite positions within the search space. Under such a simple configuration, the authors analyze difficulties that may be encountered depending on the size of the

population. However, when the problem to be solved does not represent such a trivial landscape, no theoretical methods are available to decide the optimal configuration for solving it. Thus, in real-world problems, good practice in the field suggests analyzing different parameter configurations within reasonable margins (e.g., using population sizes of the same order of magnitude as the genome size) and conducting a thorough statistical study. This process requires a significant number of runs for each configuration so that statistical tests can determine whether there are statistically significant differences between different parameter configurations (good examples about how to do it can be found for instance in De Jong (2007), Derrac et al. (2011), Eiben and Smit (2011), Karafotias, Hoogendoorn, and Eiben (2014), Rojas et al. (2002) and Qasem and Lam (2023)).

In this case, a simple-problem analysis is performed, that is, several runs of the algorithms are conducted over a given problem. In the statistical analysis, hypothesis testing is used to study inferences about one or more populations from provided samples. Two types of hypothesis are needed:  $H_0$ , the null hypothesis, which states that no difference exists (between algorithms in this case), and  $H_1$ , the alternative hypothesis, which indicates the presence of an effect. The level of significance  $\alpha$  is used to determine whether the null hypothesis can be rejected. The p-value is the probability of obtaining, by chance, a difference equal to or greater than the one observed, provided that there is no real difference in the population from which the samples come, that is, assuming that  $H_0$  is true. This value provides information about the significance of the statistical hypothesis test, and how significant is the result. Smaller p-value strengthen the evidence against  $H_0$ .

Both parametric or non parametric tests can be performed. Parametric tests assume that the sample populations have the same dispersion, meanwhile non-parametric tests do not assume a normality distribution in the data. In this paper, non-parametric tests are employed as they are

safer than parametric tests since they do not assume this normal distributions (see e.g. Derrac et al. (2011)). Non-parametric tests include two types of analyses: pairwise comparisons, which compare two algorithms and produce a p-value for each comparison; and multiple comparisons tests, which compare more than two algorithms. The non-parametric pairwise statistical procedure used in this study is the Wilcoxon signed rank test (also known as the Mann–Whitney test). This method is chosen for its simplicity, safety, and robustness. Unlike the t-test (a typical parametric procedure), it is less influenced by outliers and considers continuous differences  $d_i$ , minimizing the impact of rounding errors.

The Wilcoxon signed rank test or the Mann–Whitney test (Wilcoxon–Mann–Whitney for short) computes the differences  $d_i$  between the performance scores of both algorithms in the  $i$ th out of  $n$  problems and they are ranked according to their absolute values.  $R^+$  sums the ranks for problems in which the first algorithm gets over the second, while  $R^-$  sums the opposite. After that, the value  $T$  is computed as  $\min(R^+, R^-)$ . If  $T$  is less than or equal to the value of the distribution of Wilcoxon, the null hypothesis,  $H_0$ , is rejected. More detailed descriptions of these non-parametric statistical procedures can be found in Zar (1999) and Derrac et al. (2011).

For both APO and PAPO problems, the greedy algorithm is used initially. Due to its deterministic nature, it is enough to run this algorithm only once. However, in the case of genetic algorithms, 50 runs are performed for each experiment to obtain statistically relevant data (see e.g. Derrac et al. (2011) and Ser et al. (2019)). The best solution and the average of all solutions are reported. The purpose of running the experiments 50 times is not to find better solutions but to obtain statistically significant results, enabling the calculation of more stable mean values that are not influenced by specific runs where the inherently stochastic nature of genetic algorithms can be affected by outliers (see e.g. Suganthan et al. (2005) and Zhenyu, Jun, Zhongda, Shujiang, and Yanhong (2023)). By repeating the experiments several times, non-parametric statistical tests such as the Wilcoxon–Mann–Whitney test or the Kruskal–Wallis test can be applied to determine whether statistically significant differences exist between different configurations.

Additionally, results are compared after running the algorithm 100, 200, and 500 iterations. The objective here is to assess whether increasing execution times by adding more iterations is worthwhile or if it is better to consider a smaller number of iterations if full convergence has not yet been reached. If multiplying the runtime by five (to go from 100 to 500 iterations) only slightly improves solution quality, it may be preferable not to increase the runtime. This approach aims to gather data for informed decision-making when balancing execution time against solution quality.

## 6.2. APO experiment

As stated in Section 2.3, the first experiment consists in playing the role of a video game company, where the number of impressions to be reached in each segment is established by setting a percentage of the total number of possible impressions for that segment (*Impressions%*). The main target of the campaign will be young people, as they are the main video game consumers.

Three different scenarios are considered in the experiment to represent different objectives to be achieved. These scenarios range from a large company with a substantial budget to a smaller company capable of affording only a modest advertising campaign. Table 1 presents the specific objectives of the campaigns, where the last column (*max Impressions*) indicates the maximum number of impressions that can be achieved for each segment if advertising is done in all possible media. Thus, the minimum number of impressions to be reached for customers in each segment  $S_j$ , denoted by  $b_j \in \mathbb{N}$ , is calculated as:

$$b_j = \max Impressions(S_j) * \frac{Impressions\%}{100}$$

The experiments begin by considering only the first scenario. In this case, different configurations of the genetic algorithm are analyzed. Specifically, the results obtained with a completely random initialization of the individuals of the GA are compared to those obtained when the output of the greedy algorithm is included as an individual in the initial population. The results presented in Table 2 indicate that better average outcomes are achieved when the greedy algorithm is utilized. Furthermore, in Table 3 the results of the Wilcoxon–Mann–Whitney test, used to compare the outcomes obtained with different parameter configurations, are also shown. In particular, it can be seen that the comparison between including or not the output of the greedy algorithm yields a p-value of  $7.938E - 5$ , which is significantly lower than the  $\alpha$  value used as a threshold (0.05). This indicates a statistically significant difference in favor of including the outcome of the greedy algorithm in the initial population.

In Tables 2 and 3 the differences between various configurations are also analyzed, including population sizes, the use or non-use of elitism, different selection and crossover methods, different mutation probabilities, and different limits on the maximum number of mutations in a single individual. The results indicate that the best configuration involves using 150 individuals with elitism, 15% of mutation probability, and limiting the number of mutations to three. Statistical tests have been conducted for each comparison to evaluate the significance of the results. In particular, the Wilcoxon–Mann–Whitney test is used when only two options are compared, while the Kruskal–Wallis test is applied initially when more than two options are available. The Wilcoxon–Mann–Whitney test is then used again to compare the best configuration against each of the other configurations. Table 3 summarizes all the p-values obtained from the pairwise tests provided by Wilcoxon–Mann–Whitney test. In particular, the p-values indicate a statistically significant difference between using a population size of 150 and any of the other alternatives. They also show a statistically significant difference between using elitism and not using it. Regarding mutation rate, similar results are obtained with 15% and 12%, while the results deteriorate significantly as the mutation probability deviates from these values (both lower and higher). These findings align with the general understanding that very high mutation probabilities result in overly random searches, while very low probabilities lead to insufficient variety in the search. The experiments demonstrate that mutation levels between 12% and 15% yield very similar results (making either value interchangeable) and are better than other configurations.

For crossover methods, the experimental results suggest that uniform crossover slightly outperforms other crossover techniques. However, when the Wilcoxon–Mann–Whitney test is used for pairwise comparisons (see Table 3), a small difference is observed when compared to one-point crossover, but no statistically significant difference is found between uniform crossover and two-point crossover. In other words, uniform and two-point crossover methods are effectively indistinguishable.

For the selection method, the best results are achieved using roulette wheel selection. Moreover, the tests indicate a statistically significant difference between roulette wheel selection and truncation selection. However, no statistically significant difference is observed between roulette wheel selection and tournament selection. Thus, either of these methods can be used to achieve comparable results.

After analyzing the first scenario in detail, Table 4 summarizes the results obtained for the three scenarios. For each scenario, the results produced by the greedy algorithm and the genetic algorithm are presented. In the latter case, the results are shown with the number of iterations limited to 100, 200, and 500. Regarding the number of iterations of the GA, the results are slightly better when considering 500 iterations, though the improvement is not substantial. In fact, the algorithm’s convergence is relatively fast, making it unnecessary to consider larger numbers of iterations. The evolution of the average quality of the solutions as more iterations are performed is illustrated

**Table 1**  
Video game campaign targets for each population segment and scenario.

Population segment		Impressions%			max impressions
Age	Sex	Case 1	Case 2	Case 3	
14–19	Male	70%	50%	30%	1 163 257
	Female	70%	50%	30%	1 136 713
20–24	Male	70%	50%	30%	1 076 014
	Female	70%	50%	30%	1 008 728
25–34	Male	50%	40%	25%	2 324 988
	Female	50%	40%	25%	2 101 601
35–44	Male	35%	30%	20%	3 451 674
	Female	35%	30%	20%	2 737 366
45–54	Male	20%	20%	15%	3 847 274
	Female	20%	20%	15%	3 322 117
55–64	Male	20%	20%	15%	3 275 720
	Female	20%	20%	15%	2 971 257
65–74	Male	0%	0%	0%	2 091 688
	Female	0%	0%	0%	2 105 730
75+	Male	0%	0%	0%	1 419 362
	Female	0%	0%	0%	2 234 439

**Table 2**  
Video game campaign: Analysis of different configurations 1st scenario.

Parameters	GA 100 iterations	GA 200 iterations	GA 500 iterations	
Population size				
80	295 094.02	293 762.84	292 495.66	
100	292 182.26	291 307.48	290 782.72	
120	289 412.5	289 110.32	288 352.98	
<b>150</b>	<b>287 801.12</b>	<b>286 132.8</b>	<b>285 796.18</b>	
200	290 283.7	287 435.84	285 670.86	
250	294 051.64	286 683.18	285 813.76	
Using solution of greedy algorithm as an individual of the initial population				
Yes	<b>287 801.12</b>	<b>286 132.8</b>	<b>285 796.18</b>	
No	291 495.18	287 230.58	286 190.86	
Selection + Elitism				
<b>Roulette</b>	<b>Yes</b>	<b>287 801.12</b>	<b>286 132.8</b>	<b>285 796.18</b>
Roulette	No	290 004.01	287 148.86	286 008.58
Tournament	Yes	287 322.78	286 989.65	286 449.46
Tournament	No	292 784.48	289 190.74	287 687.9
Truncation	Yes	288 140.56	287 732.67	287 006.33
Truncation	No	299 973.16	290 876.3	290 145.12
Crossover				
One-Point	291 495.18	287 230.58	286 190.86	
Two-Point	288 069.91	287 340.75	285 953.43	
<b>Uniform</b>	<b>287 801.12</b>	<b>286 132.8</b>	<b>285 796.18</b>	
Mutation probability				
1%	290 054.3	289 871.96	289 090.56	
5%	290 070.08	289 623.78	287 847.92	
10%	289 982.32	287 328.0	287 191.36	
12%	288 102.08	286 873.66	286 104.62	
<b>15%</b>	<b>287 801.12</b>	<b>286 132.8</b>	<b>285 796.18</b>	
20%	289 514.26	287 207.74	286 169.16	
30%	290 314.34	289 324.88	286 743.94	
Maximum number of genes to be mutated				
1	290 196.2	290 112.38	288 281.1	
2	288 834.32	288 025.7	287 260.3	
<b>3</b>	<b>287 801.12</b>	<b>286 132.8</b>	<b>285 796.18</b>	
5	293 087.76	287 110.72	285 220.9	
10	300 054.54	291 243.3	287 434.56	

in Fig. 2, where a clear convergence pattern is visible starting from 200 iterations. This does not imply that the results cannot improve further with additional iterations. Indeed, it is well-known that genetic algorithms can continue improving their solutions until the optimal solution is found. However, the trade-off between computational cost

**Table 3**  
Video game campaign: Wilcoxon–Mann–Whitney Test is used (with  $\alpha = 0.05$ ) to compare the best configuration against different configurations for the 1st scenario.

Parameters	p-value
Population size 150 vs	
80	6.691e–8 < $\alpha$
100	7.017e–8 < $\alpha$
120	0.007108 < $\alpha$
200	0.02108 < $\alpha$
250	0.01954 < $\alpha$
Using solution of greedy algorithm as an individual of the initial population vs	
Not using it	7.938e–5 < $\alpha$
Roulette selection with elitism vs	
Roulette selection without elitism	3.032e–4 < $\alpha$
Tournament selection with elitism	0.6783 > $\alpha$
Tournament selection without elitism	3.452e–6 < $\alpha$
Truncation selection with elitism	0.001767 < $\alpha$
Truncation selection without elitism	1.689e–7 < $\alpha$
Uniform crossover vs	
One-Point	0.01548 < $\alpha$
Two-Point	0.4488 > $\alpha$
Mutation probability 15% vs	
1%	0.001041 < $\alpha$
5%	0.002394 < $\alpha$
10%	0.03752 < $\alpha$
12%	0.2534 > $\alpha$
20%	0.002043 < $\alpha$
30%	1.49e–4 < $\alpha$
Maximum number of genes to be mutated 3 vs	
1	2.898e–5 < $\alpha$
2	9.105e–4 < $\alpha$
5	0.006676 < $\alpha$
10	4.071e–7 < $\alpha$

and the level of improvement becomes less favorable as the number of iterations increases.

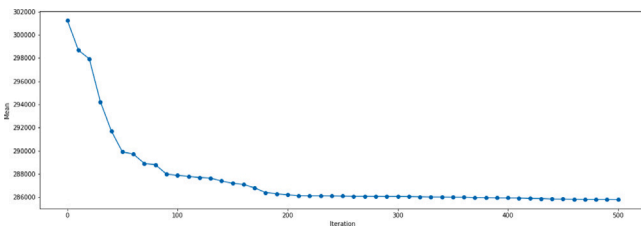
As observed, the solutions obtained by GA are never worse than those produced by the greedy algorithm (recall that the solution of the greedy algorithm is included as one of the initial individuals in the GA). Moreover, the mean improvement achieved by GA over the greedy algorithm varies across scenarios. In fact, when the number of impressions required is neither very large nor very small (second scenario), the media needed to fulfill the requirements tend to be the

**Table 4**  
Results of the video game campaigns.

Case	Greedy	GA 100 iterations		GA 200 iterations		GA 500 iterations		Mean Improvement
		Best	Mean	Best	Mean	Best	Mean	
1	301 276	284 886	287 801.12	284 814	286 132.8	284 814	285 796.18	5%
2	177 236	176 591	177 086.9	176 591	177 047.28	176 591	176 996.9	0.15%
3	102 280	88 477	95 162.54	88 477	94 070.18	88 477	93 225.78	8%

**Table 5**  
Videogame campaigns: GA vs BPSO.

Case	BPSO		GA 100 iterations		GA 200 iterations		GA 500 iterations		Mean Improve
	Best	Mean	Best	Mean	Best	Mean	Best	Mean	
1	290 149	294 591.06	284 886	287 801.12	284 814	286 132.8	284 814	285 796.18	2.8%
2	177 236	177 236	176 591	177 086.9	176 591	177 047.28	176 591	176 996.9	0.11%
3	93 991	98 228.4	88 477	95 162.54	88 477	94 070.18	88 477	93 225.78	4.33%



**Fig. 2.** Video game campaigns: Convergence evolution.

typical media targeting younger audiences. In such cases, the greedy algorithm can easily find near-optimal solutions. However, when a very high or very low number of impressions is required, GA can explore more unconventional combinations of media.

After comparing GA with the greedy method, the results of GA are compared to those obtained by another bioinspired method, specifically BPSO. To ensure a fair comparison, the run times for BPSO are set to match the run times of GA with the maximum number of iterations (500). As shown in Table 5, although BPSO also outperforms the results of the greedy algorithm, GA produces better results than BPSO. Furthermore, statistical tests confirm a statistically significant difference between GA and BPSO. This demonstrates that GA is actually better than BPSO for this problem.

Finally, the specific media most frequently selected by the GA are discussed. It should be noted that this first experiment focused on the youngest population ranges. In this context, it is unsurprising that the radio station *Los 40 Principales*, a current pop music station aligned with the target population segment, is selected in nearly all cases. The results regarding newspapers are also as expected: printed newspapers are not selected. Instead, newspaper websites such as *elpais.com*, *lavanguardia.com* and *as.com* appear consistently. For magazines, an unexpected outcome is the frequent selection of the gossip magazine *Pronto*. Although it does not seem an obvious choice for younger audiences, it proved to be a perfect fit for fulfill the required constraints in some complex combinations. Similarly, two unexpected local TV channels were selected to complete the campaigns: *Aragón TV* and *TV Canaria*. Finally, national TV channels such as *Antena 3*, *La Sexta* and *Neox* are used to achieve a large number of impressions. In particular, *Neox*, a channel focused on mainly young entertainment content, stands out as a key medium for targeting the intended audience.

### 6.3. PAPO experiment

Now, a hearing aid store is considered, and the objective is to ensure that any individual has a given probability of being exposed to the advertisement at least once. As in the example shown in the previous section, experiments are conducted with three different cases,

**Table 6**  
Hearing aids campaign targets for each population segment and scenario.

Population segment		Probability% ( $mP_i$ )			max probability%
Age	Sex	Case 1	Case 2	Case 3	
14–19	Male	0%	0%	0%	51,1%
	Female	0%	0%	0%	52,8%
20–24	Male	0%	0%	0%	55,2%
	Female	0%	0%	0%	56,3%
25–34	Male	5%	5%	0%	56,4%
	Female	5%	5%	0%	53,4%
35–44	Male	15%	10%	5%	61,4%
	Female	15%	10%	5%	52,7%
45–54	Male	25%	15%	10%	61,2%
	Female	25%	15%	10%	56,6%
55–64	Male	35%	25%	15%	61,9%
	Female	35%	25%	15%	57,7%
65–74	Male	45%	35%	25%	50,1%
	Female	45%	35%	25%	53,7%
75+	Male	45%	35%	25%	48,9%
	Female	45%	35%	25%	54,3%

representing campaigns aiming to achieve lower or higher objectives. In Table 6, the second column represents  $mP_i$ , the target probability that individuals in  $S_i$  view the advertisement at least once. The last column (*max Probability%*) represents the maximum probability that can be reached for each population segment if the advertisement is included in all possible media.

Following the same scheme as in the previous section, the experiments begin by considering only the first scenario, and different configurations of the genetic algorithm are analyzed. The average results are presented in Table 7, while the results of the statistical tests can be seen in Table 8. As in the previous campaign, better results are obtained when the solution of the greedy algorithm is included as an element of the initial population. The best population size is found to be 150 individuals, elitism is advantageous, a 15% mutation probability performs best, and limiting the number of mutations to three is optimal. Wilcoxon-Mann-Whitney or Kruskal Wallis tests are used in all cases to evaluate whether these advantages are statistically significant. The tests confirm the statistical superiority of including the outcome of the greedy algorithm and also the statistical superiority of using elitism. Regarding population size, a statistically significant difference is observed between using 150 individuals and using fewer (the difference is greater as the population is reduced). However, no significant difference is found between using 150, 200 or 250 individuals, although all other factors being equal, the smaller population size among equivalent options is preferred.

Regarding mutation rate, the situation is analogous to the previous problem. That is, the results are statistically indistinguishable when a

**Table 7**  
Hearing aids campaign: Analysis of different configurations 1st scenario.

Parameters	GA 100 iterations	GA 200 iterations	GA 500 iterations
<b>Population size</b>			
80	238 384.96	236 598.78	234 447.86
100	231 576.32	230 070.32	229 377.6
120	228 451.92	227 784.56	227 232.06
<b>150</b>	<b>226 795.06</b>	<b>225 496.72</b>	<b>224 112.26</b>
200	230 918.3	225 298.04	223 316.44
250	237 208.3	223 791.74	221 933.38
<b>Using solution of greedy algorithm as an individual of the initial population</b>			
<b>Yes</b>	<b>226 795.06</b>	<b>225 496.72</b>	<b>224 112.26</b>
No	228 603.96	227 107.24	224 369.94
<b>Selection + Elitism</b>			
<b>Roulette Yes</b>	<b>226 795.06</b>	<b>225 496.72</b>	<b>224 112.26</b>
Roulette No	228 827.86	226 586.38	226 382.28
Tournament Yes	227 132.35	224 956.12	224 769.83
Tournament No	230 002.19	229 772.74	225 931.55
Truncation Yes	226 992.56	226 809.78	225 604.05
Truncation No	232 154.9	229 897.02	226 513.33
<b>Crossover</b>			
One-Point	227 185.95	225 891.69	225 205.6
Two-Point	226 707.89	225 353.22	224 231.47
<b>Uniform</b>	<b>226 795.06</b>	<b>225 496.72</b>	<b>224 112.26</b>
<b>Mutation probability</b>			
1%	230 919.7	228 114.8	227 819.88
5%	229 868.26	228 985.94	228 463.56
10%	227 774.28	226 740.5	225 503.44
12%	228 044.34	225 432.98	225 418.74
<b>15%</b>	<b>226 795.06</b>	<b>225 496.72</b>	<b>224 112.26</b>
20%	227 848.96	226 438.06	225 438.52
30%	231 286.28	228 989.06	224 925.88
<b>Maximum number of genes to be mutated</b>			
1	229 365.52	229 359.44	228 793.22
2	227 432.9	226 910.34	226 439.34
<b>3</b>	<b>226 795.06</b>	<b>225 496.72</b>	<b>224 112.26</b>
5	229 235.5	224 881.48	223 831.74
10	231 511.18	226 822.2	224 153.62

mutation rate of 12% or 15% is used. However, statistically significant differences are observed as the mutation rate is further increased or decreased. The reason again is that very low mutation rates do not allow sufficient exploration of the search space, while excessively high mutation rates make the search overly random.

Regarding crossover and selection methods, the results are analogous to those obtained in the previous case study. First, uniform crossover seems to provide slightly better results. However, no a statistically relevant difference is observed between uniform crossover and one-point or two-point crossover, as shown in Table 8. Second, roulette wheel selection outperforms truncation selection with a statistically significant difference. However, although roulette wheel selection obtains results that seem slightly better than those of tournament selection, no statistically significant difference is found between the two methods (see Table 8).

After analyzing the first scenario in detail, Table 9 summarizes the results obtained for the three scenarios. Similar to the video game campaign, slightly better results are observed when 500 iterations are considered, though the improvement is not substantial. The algorithm's convergence is relatively fast, as shown in Fig. 3, making it unnecessary to consider larger numbers of iterations.

As expected, GA solutions are never worse than those of the greedy algorithm. Moreover, the mean improvement achieved by GA over the greedy algorithm is significantly larger in this case compared to the APO problem. Even in the simplest scenario, where typical media are enough to obtain reasonable results, GA improves the greedy algorithm's solution by an average of 4%. In the more complex scenarios, the improvements are much larger, reaching 28% and 34%.

**Table 8**  
Hearing aids campaign: Wilcoxon–Mann–Whitney Test is used (with  $\alpha = 0.05$ ) to compare the best configuration against different configurations for the 1st scenario.

Parameters	p-value
<b>Population size 150 vs</b>	
80	3.313e-7 < $\alpha$
100	3.012e-4 < $\alpha$
120	0.01227 < $\alpha$
200	0.7147 > $\alpha$
250	0.903 > $\alpha$
<b>Using solution of greedy algorithm as an individual of the initial population vs</b>	
Not using it	0.02551 < $\alpha$
<b>Roulette selection with elitism vs</b>	
Roulette selection without elitism	0.03667 < $\alpha$
Tournament selection with elitism	0.3729 > $\alpha$
Tournament selection without elitism	0.002216 < $\alpha$
Truncation selection with elitism	0.008987 < $\alpha$
Truncation selection without elitism	3.649e-5 < $\alpha$
<b>Uniform crossover vs</b>	
One-Point	0.1976 > $\alpha$
Two-Point	0.6165 > $\alpha$
<b>Mutation probability 15% vs</b>	
1%	4.108e-4 < $\alpha$
5%	0.001766 < $\alpha$
10%	0.04169 < $\alpha$
12%	0.2849 > $\alpha$
20%	0.007065 < $\alpha$
30%	2.713e-4 < $\alpha$
<b>Maximum number of genes to be mutated 3 vs</b>	
1	7.811e-4 < $\alpha$
2	0.02063 < $\alpha$
5	0.5603 > $\alpha$
10	0.01915 < $\alpha$

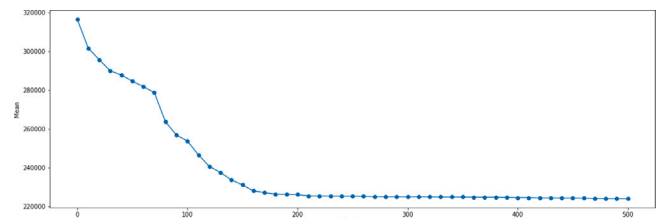


Fig. 3. Hearing aids campaigns: Convergence evolution.

This substantial improvement is attributed to the increased complexity of interrelationships between different media in the PAPO problem compared to APO. This complexity results in a combinatorial unpredictability that allows unconventional combinations of media to achieve solutions far superior to those that a greedy algorithm can provide.

Regarding the comparison between GA and BPSO, a similar situation to the previous case study is observed, as shown in Table 10. BPSO is configured to have a runtime comparable to that of GA when performing 500 iterations. The results show that BPSO significantly improves upon a greedy method. However, GA clearly outperforms BPSO, particularly in the most complex scenario. Furthermore, the Wilcoxon–Mann–Whitney test confirms a statistically significant difference between GA and BPSO.

Regarding the concrete media that have been selected, it is unsurprising that physical versions of newspapers such as ABC or Marca appear frequently, while newspaper websites are selected less often, with elpais.com being the most notable exception. For radio stations,

**Table 9**  
Results of the hearing aids campaigns.

Case	Greedy	GA 100 iterations		GA 200 iterations		GA 500 iterations		Mean Improvement
		Best	Mean	Best	Mean	Best	Mean	
1	316 754	218 344	226 795.06	218 344	225 496.72	218 344	224 112.26	28%
2	110 323	99 787	106 513.32	99 787	105 812.6	99 787	105 195.76	4%
3	74 293	47 518	49 284.36	47 518	49 008.06	47 518	48 748.96	34%

**Table 10**  
Hearing aids campaigns: GA vs BPSO.

Case	PSO		GA 100 iterations		GA 200 iterations		GA 500 iterations		Mean Improve
	Best	Mean	Best	Mean	Best	Mean	Best	Mean	
1	226 149	237 407.2	218 344	226 795.06	218 344	225 496.72	218 344	224 112.26	5.3%
2	108 514	109 364.52	99 787	106 513.32	99 787	105 812.6	99 787	105 195.76	3.33%
3	51 681	57 227.8	47 518	49 284.36	47 518	49 008.06	47 518	48 748.96	16.76%

music-focused stations are not selected, whereas general-purpose radio stations such as *SER* and *COPE* do appear. Similar to the case of video games campaigns, national TV channels are required to reach larger populations. Additionally, it is notable that local TV channels are selected much more frequently in this case, particularly: *Canal Sur*, *TV de Galicia*, *Aragón TV*, and *TV Canaria*.

6.4. Discussion

In the problems under consideration, non-parametric tests have proved that the use of genetic algorithms provides statistically significant advantages over other techniques, such as greedy algorithms or Particle Swarm Optimization. However, it has also been shown that genetic algorithms achieve better results when one of the elements of the initial population is provided by a greedy algorithm. The reason of this advantage lies in the approximability class to which these problems belong. As highlighted in [Muñoz and Rubio \(2021\)](#), genetic algorithms become more competitive as the problems to be solved become more difficult to approximate. Since the problems addressed in this paper are Log-APX-hard, they are situated in the ideal range for the use of genetic algorithms supported by a greedy algorithm. If the approximability class were better (e.g. FPTAS), then genetic algorithms would provide less advantage. Conversely, if the approximability class were worse (such as Exp-APX), genetic algorithms would not require the support from greedy algorithms.

Moreover, when the differences between the APO and PAPO case studies are considered, it has been found that the advantage of genetic algorithms over the greedy algorithm is greater in PAPO than in APO. This is due to the higher complexity of the PAPO problem. In PAPO, a larger amount of information must be managed, and, more importantly, a higher level of constraints must be addressed: since probabilistic interrelationships must be considered, the casuistry is much higher. This increased complexity further emphasizes the superiority of the search capabilities of genetic algorithms compared to a greedy algorithm.

7. Conclusions and future work

In this paper, two ad positioning optimization problems have been formally defined. The first one (APO) aims to ensuring a minimum number of impressions (views) in each population segment. The second one (PAPO) focuses on ensuring a minimum probability for each population segment that any individual watches the ad at least once, taking into account the dependencies between the audiences of different media.

Greedy algorithms and also genetic algorithms have been provided to solve these problems, and their usefulness has been evaluated by considering two specific advertisement campaigns using real data. Additionally, an implementation using a binary version of particle swarm optimization (BPSO) has been presented. In the case of APO, the results

show that the genetic algorithm improves the quality of the solutions obtained by both the greedy algorithm and BPSO, specially in the most complex scenarios. However, the improvement is not as significant as in the case of PAPO. For PAPO, the genetic algorithm produces results that are much better than those of the greedy algorithm and also clearly outperforms BPSO.

Beyond the two case studies presented (video games and hearing aids), two other case studies have been developed but are not included in this paper to avoid excessive length. Specifically, a wedding dress campaign (targeting primarily women of marriageable age) and a hair implant campaign (targeting primarily middle-aged men) have been analyzed. The results from these cases are consistent with those obtained with video games and hearing aids. That is, genetic algorithms achieve much larger advantages in the case of PAPO than in APO, though they outperform the greedy algorithms in both cases.

As future work, efforts will be made to improve the expressiveness of the dependency constraints used to define PAPO instances. This could involve defining new types of dependencies (between independence and exclusion, between independence and inclusion). These new dependencies would allow for the creation of more precise constraints between the audiences of different media, potentially improving the results.

One of the main challenges encountered was related to the data used for testing. To define the costs per advertisement in the different media, an attempt was made to obtain the Advertising Rates from a period as close as possible to the year 2021, between September and December, as the probabilities and impressions used were collected in this interval. However, obtaining more updated data remains a goal, and efforts are underway to acquire this information.

In addition to the previous mentioned aspects, another key aspect to be addressed as future work is the development of new variants of the heuristic algorithms presented in this paper. In particular, the possibility of developing hybrid algorithms that combine BPSO and genetic algorithms, while also retaining the auxiliary use of greedy algorithms, is being studied. Although genetic algorithms have demonstrated their superiority over the other two strategies, it is possible that the creation of heuristics combining different strategies simultaneously may yield slightly better results in certain circumstances. More specifically, it would be possible for both algorithms to be run in parallel, with periodic exchanges of information about the solutions obtained by each of them. By doing this, even if BPSO in general does not independently find better solutions, it could enable the genetic algorithm to perform a broader exploration of the search space, helping to avoid local optima. Although a slower run time would be expected, this approach might be worth exploring if it achieves significant improvements in results.

CRedit authorship contribution statement

**Álvaro Seco:** Conceptualization, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft,

Writing – review & editing, Visualization. **Natalia López:** Conceptualization, Methodology, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Supervision. **Fernando Rubio:** Conceptualization, Methodology, Formal analysis, Investigation, Resources, Writing – original draft, Writing – review & editing, Supervision, Project administration, Funding acquisition.

### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

### Acknowledgments

Work partially supported by projects PID2019-108528RB-C22, PID2023-149943OB-I00, and by Comunidad de Madrid as part of the program S2018/TCS-4339 (BLOQUES-CM) co-funded by EIE Funds of the European Union. The authors would like to thank the anonymous reviewers for valuable suggestions on a previous version of the paper. They would also like to thank Ismael Rodríguez for valuable discussions and language corrections.

### Data availability

Data will be made available on request.

### References

- AIMC (2022). What is EGM? <https://www.aimc.es/en/what-we-do/>. (Accessed 6 October 2022).
- Arora, S., & Thota, S. R. (2024). Using artificial intelligence with big data analytics for targeted marketing campaigns. *International Journal of Advanced Research in Science, Communication and Technology (IJARSTC)*, 4(3), 593–602.
- Balducci, B., & Marinova, D. (2018). Unstructured data in marketing. *Journal of the Academy of Marketing Science*, 46(4), 557–590.
- Barajas, J., Kwon, J., Akella, R., Flores, A., Holtan, M., & Andrei, V. (2012). Marketing campaign evaluation in targeted display advertising. In *Proceedings of the sixth international workshop on data mining for online advertising and internet economy* (pp. 1–7).
- Brandt, F., Conitzer, V., Endriss, U., Lang, J., & Procaccia, A. D. (2016). *Handbook of computational social choice*. Cambridge University Press.
- Buchfuhrer, D., Schapira, M., & Singer, Y. (2010). Computation and incentives in combinatorial public projects. In *Proceedings of the 11th ACM conference on electronic commerce* (pp. 33–42). ACM.
- Chan, C. C. H., Hwang, Y.-R., & Wu, H.-C. (2016). Marketing segmentation using the particle swarm optimization algorithm: a case study. *Journal of Ambient Intelligence and Humanized Computing*, 7, 855–863.
- Chen, W., Li, F., Lin, T., & Rubinstein, A. (2015). Combining traditional marketing and viral marketing with amphibious influence maximization. In *Proceedings of the sixteenth ACM conference on economics and computation* (pp. 779–796).
- Chen, T., Tang, K., Chen, G., & Yao, X. (2012). A large population size can be unhelpful in evolutionary algorithms. *Theoretical Computer Science*, 436, 54–70.
- Chiarandini, M., Basso, D., & Stützel, T. (2005). Statistical methods for the comparison of stochastic optimizers. In *MIC2005: the sixth metaheuristics international conference* (pp. 189–196). Vienna Austria.
- Chiong, R., Fan, Z., Hu, Z., Adam, M. T., Lutz, B., & Neumann, D. (2018). A sentiment analysis-based machine learning approach for financial market prediction via news disclosures. In *Proceedings of the genetic and evolutionary computation conference companion* (pp. 278–279).
- Chiong, R., Fan, Z., Hu, Z., & Dhakal, S. (2022). A novel ensemble learning approach for stock market prediction based on sentiment analysis and the sliding window method. *IEEE Transactions on Computational Social Systems*, 10(5).
- Chvátal, V. (1979). A greedy heuristic for the set covering problem. *Mathematics of Operations Research*, 4, 233–235.
- Cura, T. (2023). Hybridizing local searching with genetic algorithms for the job sequencing and tool switching problem with non-identical parallel machines. *Expert Systems with Applications*, 223, Article 119908.
- Curtis, S. A. (2003). The classification of greedy algorithms. *Science of Computer Programming*, 49(1–3), 125–157.
- Das, A., Yadav, L., Singhal, M., Sachan, R., Goyal, H., Taparia, K., et al. (2016). Smart glove for sign language communications. In *2016 international conference on accessibility to digital world* (pp. 27–31). IEEE.
- DBI Data Business Intelligence (2020). All you need to know about the famous GRP. <https://dbibyhasvas.io/es/blog/>. (Accessed 29 May 2022).
- De Jong, K. (2007). Parameter setting in EAs: a 30 year perspective. In *Parameter setting in evolutionary algorithms* (pp. 1–18). Springer.
- De Jong, K. A., Spears, W. M., et al. (1989). Using genetic algorithms to solve NP-complete problems. In *ICGA* (pp. 124–132).
- Degterev, A. (2019). *Algorithmic marketing as a replacement for traditional marketing research* (Ph.D. thesis), Metropolia University of Applied Sciences.
- Del Ser, J., Osaba, E., Molina, D., Yang, X.-S., Salcedo-Sanz, S., Camacho, D., et al. (2019). Bio-inspired computation: Where we stand and what's next. *Swarm and Evolutionary Computation*, 48, 220–250.
- Derrac, J., García, S., Molina, D., & Herrera, F. (2011). A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation*, 1(1), 3–18.
- Dinur, I., & Steurer, D. (2014). Analytical approach to parallel repetition. In *Proceedings of the forty-sixth annual ACM symposium on theory of computing* (pp. 624–633).
- Eiben, A. E., & Smit, S. K. (2011). Parameter tuning for configuring and analyzing evolutionary algorithms. *Swarm and Evolutionary Computation*, 1(1), 19–31.
- Fulgioni, G. M. (2015). Is the GRP really dead in a cross-platform ecosystem?: Why the gross rating point metric should thrive in today's fragmented media world. *Journal of Advertising Research*, 55(4), 358–361.
- Gad, A. G. (2022). Particle swarm optimization algorithm and its applications: a systematic review. *Archives of Computational Methods in Engineering*, 29(5), 2531–2561.
- Han, X. (2020). A comparative study of big data marketing and traditional marketing in the age of internet. vol. 1574, In *Journal of physics: conference series*. IOP Publishing, Article 012038, 1.
- Hicham, N., & Karim, S. (2022). Machine learning and marketing campaign: Innovative approaches and creative techniques for increasing efficiency and profit. In *The international conference of advanced computing and informatics* (pp. 40–52). Springer.
- Hurley, S., Moutinho, L., & Witt, S. F. (1998). Genetic algorithms for tourism marketing. *Annals of Tourism Research*, 25(2), 498–514.
- Jin, W., Hu, Z., Chan, C., et al. (2017). An innovative genetic algorithms-based inexact non-linear programming problem solving method. *Journal of Environmental Protection*, 8(03), 231.
- Karafotias, G., Hoogendoorn, M., & Eiben, Á. E. (2014). Parameter control in evolutionary algorithms: Trends and challenges. *IEEE Transactions on Evolutionary Computation*, 19(2), 167–187.
- Katoch, S., Chauhan, S. S., & Kumar, V. (2021). A review on genetic algorithm: past, present, and future. *Multimedia Tools and Applications*, 80(5), 8091–8126.
- Keller, O., Hassidim, A., & Hazon, N. (2019). New approximations for coalitional manipulation in scoring rules. *Journal of Artificial Intelligence Research*, 64, 109–145.
- Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. vol. 4, In *Proceedings of ICNN'95-international conference on neural networks* (pp. 1942–1948). IEEE.
- Lewenberg, Y., Lev, O., & Rosenschein, J. S. (2017). Divide and conquer: Using geographic manipulation to win district-based elections. In *Proceedings of the 16th conference on autonomous agents and multiAgent systems* (pp. 624–632).
- Lilien, G. L., & Rangaswamy, A. (2004). *Marketing engineering: Computer-assisted marketing analysis and planning*. DecisionPro.
- Liu, H.-H., & Ong, C.-S. (2008). Variable selection in clustering for marketing segmentation using genetic algorithms. *Expert Systems with Applications*, 34(1), 502–510.
- Lobo, F. G., & Lima, C. F. (2005). A review of adaptive population sizing schemes in genetic algorithms. In *Proceedings of the 7th annual workshop on genetic and evolutionary computation* (pp. 228–234).
- Maymin, P. (2011). Markets are efficient if and only if P=NP. *Algorithmic Finance*, [ISSN: 2158-5571] 1(1), 1–11.
- Miralles-Pechuán, L., Ponce, H., & Martínez-Villaseñor, L. (2018). A novel methodology for optimizing display advertising campaigns using genetic algorithms. *Electronic Commerce Research and Applications*, 27, 39–51.
- Mitchell, M. (1998). *An introduction to genetic algorithms*. MIT Press.
- Muñoz, A., & Rubio, F. (2021). Evaluating genetic algorithms through the approximability hierarchy. *Journal of Computer Science*, 53, Article 101388.
- Nuara, A. (2020). *Machine learning algorithms for the optimization of internet advertising campaigns* (Ph.D. thesis), Politecnico di Milano.
- ODEC (2022). TOM micro. <https://www.odec.es/es/medios-publicitarios/>. (Accessed 25 October 2022).
- Olympia, R., Stefka, F., & Paprzycki, M. (2013). Influence of the population size on the genetic algorithm performance in case of cultivation process modelling. In *Federated conference on computer science and information systems* (pp. 371–376).
- Pedersen, M. E. H. (2010). *Good parameters for particle swarm optimization*. Tech. rep. HL1001, (pp. 1551–3203). Copenhagen, Denmark: Hvass Lab.
- Piotrowski, A. P., Napiorkowski, J. J., & Piotrowska, A. E. (2020). Population size in particle swarm optimization. *Swarm and Evolutionary Computation*, 58, Article 100718.
- Qasem, A. G., & Lam, S. S. (2023). Prediction of wart treatment response using a hybrid GA-ensemble learning approach. *Expert Systems with Applications*, 221, Article 119737.
- Robles, J. F., Chica, M., & Córdón, Ó. (2016). Incorporating awareness and genetic-based viral marketing strategies to a consumer behavior model. In *2016 IEEE congress on evolutionary computation* (pp. 5178–5185). IEEE.

- Robles, J. F., Chica, M., & Cordón, O. (2020). Evolutionary multiobjective optimization to target social network influentials in viral marketing. *Expert Systems with Applications*, 147, Article 113183.
- Rodríguez, I., Rabanal, P., & Rubio, F. (2017). How to make a best-seller: Optimal product design problems. *Applied Soft Computing*, 55, 178–196.
- Rodríguez, I., Rubio, F., & Rabanal, P. (2016). Automatic media planning: optimal advertisement placement problems. In *2016 IEEE congress on evolutionary computation* (pp. 5170–5177). IEEE.
- Rojas, I., González, J., Pomares, H., Merelo, J., Castillo, P., & Romero, G. (2002). Statistical analysis of the main parameters involved in the design of a genetic algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 32(1), 31–37.
- Serrano, A. L., & Blum, C. (2022). A biased random key genetic algorithm applied to target set selection in viral marketing. In *Proceedings of the genetic and evolutionary computation conference* (pp. 241–250).
- Solomon, S., & Uzrad, A. (2023). Dynamic  $(1 + \epsilon) \ln n$ -approximation algorithms for minimum set cover and dominating set. In *Proceedings of the 55th annual ACM symposium on theory of computing* (pp. 1187–1200).
- Souren, R., Ahn, H., & Schmitz, C. (2005). Optimal product mix decisions based on the theory of constraints? Exposing rarely emphasized premises of throughput accounting. *International Journal of Production Research*, 43(2), 361–374.
- Suganthan, P. N., Hansen, N., Liang, J. J., Deb, K., Chen, Y.-P., Auger, A., et al. (2005). Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. *KanGAL Report, 2005005*(2005), 2005.
- Vazirani, V. V. (2001). *Approximation algorithms: vol. 1*, Springer.
- Waqas, M. (2020). *Feature generation for optimization of marketing campaign* (Master's thesis), M. Waqas.
- Wedel, M., & Kamakura, W. A. (2000). Market segmentation: Conceptual and methodological foundations. In *International series in quantitative marketing: vol. 8*, Springer.
- Yun, J. T., Segijn, C. M., Pearson, S., Malthouse, E. C., Konstan, J. A., & Shankar, V. (2020). Challenges and future directions of computational advertising measurement systems. *Journal of Advertising*, 49(4), 446–458.
- Yunzhi, W., Yaoguang, H., Yanhua, J., & Ruijun, Z. (2010). A game theoretic approach to product-mix resource allocation. In *IEEE conference on industrial electronics and applications* (pp. 2142–2147). IEEE.
- Zar, J. H. (1999). *Biostatistical analysis*. Pearson Education India.
- Zehmakan, A. N., Zhou, X., & Zhang, Z. (2023). Viral marketing in social networks with competing products. arXiv:2312.15819. URL: <https://arxiv.org/abs/2312.15819>.
- Zhenyu, K., Jun, Z., Zhongda, T., Shujiang, L., & Yanhong, W. (2023). NSGAIII based on utopian point improvements and its application in wastewater treatment process. *Expert Systems with Applications*, 234, Article 121030.