OBSERVING AND OPTIMIZING ONLINE AD ASSIGNMENTS

BY DARJA KRUŠEVSKAJA

A dissertation submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Graduate Program in Computer Science

Written under the direction of

S. Muthukrishnan

and approved by

New Brunswick, New Jersey

October, 2015

ABSTRACT OF THE DISSERTATION

Observing and Optimizing Online Ad Assignments

by Darja Kruševskaja

Dissertation Director: S. Muthukrishnan

The main focus of this thesis work is on optimization and observation of ad assignments in online ad markets. Online ad markets allocate billions of impressions to advertisers while satisfying an array of constraints. Their revenues support the Internet ecosystem. They highlight theory problems and inspire systems research. In this thesis work we initiate the study that seeks to understand mechanisms and dynamics of advertising markets. We develop a scalable crawling capability that allows us to harvest a corpus of ads across a large number of websites and user profiles. We establish that user profile is essential in display ad markets: 50% of observed websites have at least 80% of their ads targeted at profiles. Further, we introduce *cardinal auctions* for selling multiple copies of a good, in which bidders specify not only their bid or how much they are willing to pay for the good, but also a cardinality constraint on the maximum size of the allocation in which they are willing to participate. We perform the first known analyses of Price of Anarchy and revenue of cardinal auctions. Finally, we introduce a new class of online allocation problems with *secondary metrics*, in which the goal is to optimize one metric (*e.g.,* revenue), while meeting another (*e.g.,* cost of user conversion). We suggest a number of theoretical approaches to the problem and test one of them in a real-world setting by using it in ad allocation in a ad network.

Acknowledgements

This thesis is the result of hard work. People around me were a necessary component for its completion.

I am forever grateful to my adviser Muthu (S.Muthukrishnan). His elegant style of thinking and reasoning is the reason I have learned so much about algorithms. His graduate course on advanced algorithms was the main reason why I work in the area of online ad markets. Hours of conversations with him shaped me not only as a researcher, but also as a person. Willingly or not I inherited his view of the world. He has been a great support throughout the years. His patience with me is remarkable. I am honored to be his student.

I would like to thank my committee members Muthu, Martin, Badri and Yaron for taking time to serve on my committee.

I thank my internship mentor at Google Research, Mark Sandler. His smartness is an inspiration. He continued to be my mentor after my internship was over. I thank William Simpson for being my mentor at Flurry. His natural way of reasoning and navigating through very complex problems had a great impact on me and my approach to work.

I am honored to be a part of Massdal Lab, where I met a lot of great students and colleagues. Qiang Ma, Vasisht Gopalan, Edan Harel, my life would not be quite the same without you! I am grateful to Priya Govindan for being the *Godmother* of my thesis.

Many thanks to co-authors and collaborators: Paul Barford, Igor Canadi, Mangesh Gupte and Qiang Ma. I am grateful for the conversations and work with Michael Saks and Stratis Ioannidis. I have learned a lot from my work with Vladimir Pavlovic.

To support the argument that I took after Muthu, I would like to thank couple of

“non CS" people. I am grateful to Amy Farid, my hairdresser, for introducing me to TM. I am lucky to have Dick Talens as my fitness coach. He taught me about hard work, dedication and self-compassion. He proved me that nothing is impossible if one just sticks with the plan. I am grateful to Mary Burns, in a very hard moment she helped me and reminded me that sometimes people just do wonderful things for each other.

I want to thank my friends. They have influenced me greatly: Dan Bodganov, Liina Kamm, Pavel Grigorenko, Jevgeni Kabanov, Natalya Myachina, Sergey Andreev, Hedi Peterson, Konstantin Tretyakov, Katrin Tsepelina, Mark Kofman. Each of you gave me a spark, a conversation, a thought, a support.

Finally, there is my family. But how can one quantify their role? I am thankful to my immediate family: my mom, dad and my sister Asya. They have raised me, played with me, challenged me, and pushed me. My mom allowed me to pursue my dreams and was always there for me when I needed her. My dad, being the smart person that he is, kept getting my sister and I toys like Rubik’s cube. I also grateful to my in-law family. Their love, patience and support were always there for me. My extended family supported me on this continent, especially in the beginning. Inna, Yan, Shifra, Ari, Daniela, Sasha, I thank you for your support.

I am beyond grateful to my husband, Anton Litvinenko. He was and is much more than a husband. He taught me programming and algorithmic thinking. He believed in me, at times more than I did myself. He never allowed me to settle for the average, and always encouraged me to pursue my goals. His support was to push me every single day. I am thankful for your patience, love, support and encouragement. I will be forever baking brownies for you!

Thank you! Aitäh! Спасибо!

Dedication

To my parents:

both biological and academic

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Online ad markets play an important role in the Internet ecosystem. Brands and e-commerce vendors seeking engagement with potential customers, as well as online *publishers* (*e.g.,* websites or mobile applications) searching to maintain their existence through advertising revenue, find online advertising a compelling proposition. Online advertisements, or *ads*, are sold in *online ad markets*. The online market is a virtual space that matches user visits to publisher properties with advertisers. A state-of-the-art online ad market is a service that may be called by participants at any point in time. There is a large number of markets. The dynamics and properties of each individual market differ depending on the particulars of that market. We discuss the ecosystem of online ad markets in more details in Chapter 2.

The ubiquity of advertising in publisher web sites and mobile applications makes it easy to overlook the diversity and complexity of the online ad-delivery ecosystem. A well-known depiction of the online ad ecosystem is the *Display Lumascape* (see Figure 2.1) that shows marketers (brands) and publishers/consumers connected by hundreds of companies that provide a variety of intermediary services. According to an IAB press release[1] in 2014 Internet ad revenues totaled $42.8 billion, surpassing broadcast television for the first time. Online ad markets operate at a very high rate. Google alone processes on average over 40,000 search queries every second[2]. This translates to over 3.5 billion searches per day and 1.2 trillion searches per year worldwide. Most search results are accompanied by *search ads*. Other types of ads can be seen on a variety of

[1] http://www.iab.net/media/file/IAB_Internet_Advertising_Revenue_FY_2014.pdf

[2] http://www.internetlivestats.com/google-search-statistics/

websites and in mobile applications. Ads are often selected specifically for each invocation of a publisher, or *impression*. Ads are blended into the design and content of the page on which they appear. Sometimes it is difficult to distinguish ads from content, even for a human user. Understanding the properties, dynamics and mechanisms of the market is a fundamental challenge in seeking better performing markets and a better user experience.

While there is a body of research studying revenue optimal mechanisms for ad selection at an impression level [14, 48, 42], or optimal matching problems at the market level (see [38] for an excellent survey), there is a limited understanding of how these markets transpire in practice. We initiate a study that seeks to broadly understand the features, mechanisms and dynamics present in online ad markets. We define a function *adscape* that captures the mapping from the impression to ads displayed to the user. We perform an empirical study that investigates the properties of the adscape for display and video ad markets. Our study takes the perspective of users who are the targets of ads shown on web sites. We develop a scalable crawling capability, at the heart of which is the notion of *profile-based crawling*, that enables each crawler instance to interact with the ad ecosystem as though it were a unique user with specific characteristics. We deploy our crawler over a variety of websites and profiles and harvest ads, which we then analyze. We make and report a number of observations [8].

In the adscape study we observe that the number of ads, even on a single page, can vary from one visit of the user to another. Consider display ads, or visual ads, on a webpage. Advertisers whose ads are shown on the page compete for attention of the viewers. Clearly, the number of ads shown is an important feature and publishers recognize that showing fewer ads helps[3]. Currently, this cardinality is largely determined by the publisher of the web page, who may choose to make it exclusive, showing only one ad, but in many cases will mix several. The publisher chooses the number of ads on a page based on variety of techniques, from machine learning to user studies, esthetics of UI design and revenue maximization. This approach does not let advertisers influence

[3] `http://www.technologyreview.com/news/419897/fewer-ads-more-clicks/`

how many other ads appear with their own; hence, they bid depending on the average of their values over the possible number of ads that might appear on that page. This induces inefficiencies and potential revenue loss. We study *cardinal auctions*, which allow advertisers to explicitly specify how many other advertisers may appear with their ad on a given page. We then perform a theoretical analysis of the auction [26].

The online market can be modeled as a bipartite graph. Each ad is a node in the graph, as is each impression. Online advertising constitutes perhaps the largest matching problem in the world, both in terms of dollars and number of items. The problems that arise in online ad markets have generated a lot of interest in the algorithm community. A large number of problems, models and techniques were inspired by online ad markets. This is not only due to the importance of motivation, but also due to the new and elegant questions and techniques that emerge. There has been significant progress in online matching topics as applied to more established ad markets, such as display ads ([18, 30, 31]) or search ads ([39, 27]). We observe the in-app install business market and notice that advertisers in this market are bidding and paying per *click*, however they have a maximum amount they are willing to pay per *conversion*. Inspired by this market, we propose a new class of online matching problems with *secondary targets*. The goal of the matching is not only to maximize one metric (*e.g.,* number of clicks or total revenue), but also to satisfy a *secondary metric* (*e.g.,* number of conversions or conversion average cost). We propose a number of theoretical approaches to solving the problem. None of these approaches fully captures the problem. Moreover, in practice none of the presented theoretical approaches guarantees to produce a feasible solution. Therefore, we propose and evaluate in a real-world setting two ranking-based and a Knapsack-based mechanisms and consider the results.

Chapter 2

Overview of Online Ads Ecosystem

Online advertising takes on a variety of forms, from text snippets displayed alongside search results to large flash videos overlaying the whole page. Ads can be finely tailored for a particular user based on many factors. For instance, the ad may depend on the physical location of the user or the webpage that hosts the ad. It can also depend on the user's properties, such as demographics, and the user's interests or other factors, such as market conditions. The *allocation*, or selection, of an ad and its pricing often happens in real time on a *per-impression basis* via auctions. The simplest transaction begins with the user visiting the website. The visit triggers an *ad request* that can include various pieces of information about the webpage or the user. The request goes from the website, often also referred to as the *publisher*, to an ad network. The ad network runs an auction, in which a set of ads that are eligible for the impression compete in the auction. The winning ad or ads are chosen and priced accordingly. Once the winning ads are chosen, they are plugged into the website. This transaction happens in milliseconds, while the webpage is loading at the user's end.

What seems to be a simple transaction varies depending on the website that hosts the slot for the ad. The website may choose to monetize its users on its own or use the services of intermediaries such as an ad network or an ad exchange (introduced in more details in Section 2.1). In this chapter we introduce the ecosystem of online advertising. We will consider different types of ads and ad markets and discuss a variety of roles and strategies that arise in the market. We will end the chapter by laying out different avenues of research.

2.1 Roles in the Market

Originally, websites were monetizing their users on their own. The advertising contracts were made between the website or the *publisher* and *advertisers.* The signing of contracts was both time and labor intensive; sales representatives of both sides had to meet personally. The contract would usually include the number of *impressions* (*i.e.,* number of times the publisher had to show the advertiser's ad) and the payment. The contract would also describe the penalty, for instance monetary; the publisher would have to pay to the advertiser if a lesser number of impressions was shown.

With the growth of the Internet and the development of online advertising this approach did not scale. In addition, the online advertising offered a unique proposition of targeting particular users and not crowds; which the old-fashioned contracts did not provide. Following are some of the challenges that result from a sophisticated structure of the market. There are tens of thousands of advertisers and millions of publishers. The target audience of a given advertiser can visit a variety of websites. In the old-fashioned scenario, in order to reach the target audience, the advertisers would have to have a contract with each particular publisher. However, a stand-alone publisher would only have the information on the activity of the users on their website, which is inadequate to learn about the user's interests and identify the target audience. Therefore, there has been a shift towards a more complex process that involves more companies, teams and technologies. Today advertisers and publishers are connected by hundreds of companies that provide a variety of intermediary services. There are companies that provide information on users as a service (*e.g.,* Datalogic). Others help advertisers find users that have visited their website earlier (*e.g.,* Adroll). The ecosystem of online advertising naturally divides itself into overlapping markets: search engine marketing, display advertising, social advertising, etc. Each has its unique features, properties and opportunities which we discuss later in Section 2.2.

The ecosystem of a market is often described using *Lumascapes.* Figure 2.1 shows Lumascape of display advertising. It captures roles (discussed in Section 2.1) together with companies that fill these roles in the market. Following the arrows in the figure, one can trace a variety of ways an advertiser can get his ad to the publisher's page. It is important to notice that not all steps are necessary. For instance, the advertiser can access the market through an ad network or demand side platform (DSP).

The appearance of so many different companies can be explained by the race to make all the complexity of ad impression buying disappear. For instance, the advertiser can create an agreement with an ad network, then the ad network would find the best impressions accessible to it to satisfy the agreement. Similarly, publishers can build an automated pipeline that could call an ad network upon user visits and show the ad, selected by the ad network, to the user. The core of this world is captured in Figure 2.2. There are two main sides to the diagram: *advertisers* (marketers) that form the demand on the market, and *publishers* that create the supply. And there are a number of intermediary companies that make the per-impression allocation of ads possible. This simplified representation does not include all the roles in the ecosystem. For instance, *data brokers* (who play a key role in data collection) and user profiling are not included.

While there are many roles on the market, here we only discuss the most important ones.

Publishers. We refer to websites of mobile applications that display ads along with organic content as *publishers*. Frequently advertising is the primary source of revenue for publishers. A publisher produces content or provides services that attract users and thus creates opportunities to present ads to those users. The opportunities that realize are *impressions*. For example, Google is a publisher because it provides the service of web search and presents ads to its users when they use this service. Facebook is also a publisher. It displays ads to its users when they are browsing their social network account. Publishers can choose from a variety of ad formats, payment schemes and strategies to incorporate ads. In addition, on any given visit, ads can be served from a variety of sources.

1. The publisher's sales team could set up contracts with advertisers to display ads (impressions), and ads from any of those contracts may be shown. Publishers

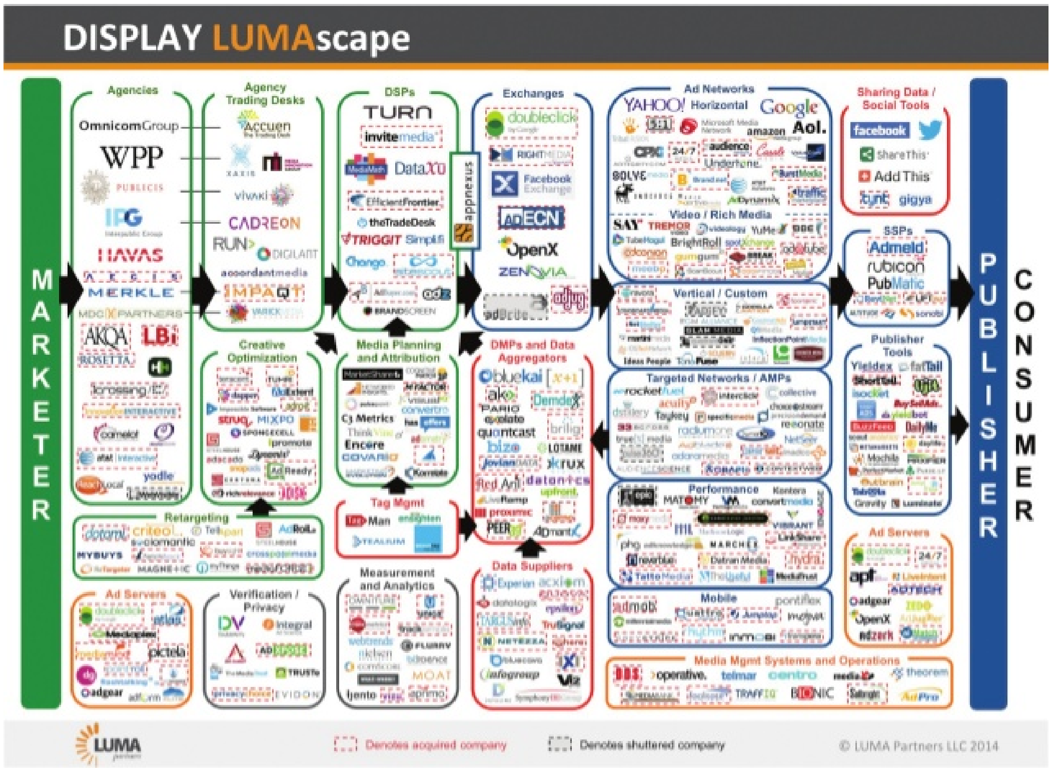


Figure 2.1: Lumascape for display advertising market. The ecosystem of display ad market is often described using Lumascape

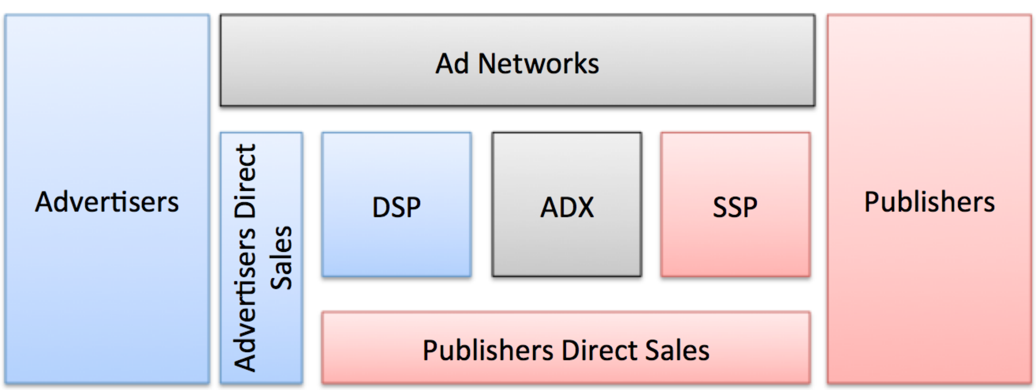


Figure 2.2: Sketch of main roles in the online advertising market.

may have information on their users (demographics, interests) and can obtain other necessary information (profiles, cookie-based targeting) from third parties as needed to match users to the targeting criteria of advertisers.

2. The publisher could outsource ad placement to one or more ad networks that represent multiple advertisers. The ad networks would choose ads to show based on publisher-specified preferences and controls.

3. The publisher can access an ad exchange and auction an ad placement opportunity in real-time. When the ad exchange receives the ad request from the publisher, it forwards it to number of ad networks and solicits bids for it. Once ad exchange gets bids it usually runs an auction. The winning ad network places the ad of its choice. In addition to sources of information above, an ad exchange may enable the sale and purchase of third-party sources of user data. This data can then be used in real-time to enhance targeted advertising.

Typically, publishers combine the methods, even on a single page.

Advertisers. Companies have been advertising their services and products for centuries. They design ad campaigns to attract the attention of potential customers. However, design of online ad campaigns may involve more details. Online ad campaigns may include creatives, target demographics, placements, frequency caps, *etc.* All of these considerations must be balanced with an advertising budget. Following are some examples of ad campaigns:

1. *A company that produces running shoes targets users who live in Madison, WI, from 6AM to 9AM on business days, and the ad must not be shown to the same user more than 7 times on any given day.* This campaign targets users based on location ("geo" in ads parlance), time of the day (one or more "dayparts"), and further limits the number of times the ad may be shown in a period of time ("frequency cap");

2. *A company that produces running shoes targets male users aged* $[25 - 30]$ *who are interested in "sports," "healthy lifestyle" and "jogging."* This campaign targets users

Figure 2.3: Example of untargeted vs. targeted ads. On the left is the first online banner ad, published on October 25, 1994. On the right is an ad that was shown to *health - mental health* oriented browser profiles recently.

based on their interests ("profile" in advertising parlance) and also demographics (gender, age).

3. *The company* `Nike` *that produces running shoes targets users who have* previously *visited* ***http://nike.com***. This is known as "retargeting." This would typically be implemented via cookies and cookie-matching methods[1].

4. *A company that produces running shoes targets users who have interests similar to other users who have recently visited* ***http://nytimes.com/sports***. This is known as "look-alike" targeting in ad parlance, and is typically implemented by learning users' profiles.

Targeting strategies can be combined in sophisticated ways by advertisers, and the industry relies on the existence of players who can track cookies and maintain user profiles.

Supply-Side Platforms. There are millions of publishers on the Internet. Some larger publishers (*e.g.,* Google, The New York Times) are able to find buyers for their impressions without intermediaries. Smaller publishers may need some help finding advertisers to fill the impressions. Supply-side platforms (SSP's) allow publishers to connect their inventory to the demand side of the market, by calling multiple ad exchanges, demand-side platforms and networks at once. This in turn allows a range of potential buyers to have access to a range of impressions coming from different publishers. The idea is that by opening up impressions to as many potential buyers as possible, often through real-time auctions, publishers can maximize the revenues they receive for their inventory. When an SSP throws impressions into ad exchanges, buyers (for instance, demand-side

[1]For example, see `https://developers.google.com/ad-exchange/rtb/cookie-guide`

platforms discussed below) analyze and purchase them on behalf of marketers depending on certain attributes (such as where they are served), and which specific users they are being served to. Sometimes they also allow publishers to set up private markets, *i.e.,* create a closed bidding process for the premium inventory. SellerCloud of Rubicon project[2] is an SSP that helps publishers such as ABC, Forbes, Time Inc., The Tribune Company and Virgin Media to find good advertisers for their impressions.

Demand-Side Platforms. In 2014, the top 10 of advertisers in the USA has spent $15.34 billion dollars on traditional media and online display advertising[3]. Large spenders can afford to have large sales teams, and important publishers, such as The New York Times, are likely to sign contracts with them directly. However, there are millions of smaller advertisers. Smaller advertisers can access online ad markets through demand-side platforms (DSP's). DSP's allow advertisers' demand to access a wide range of supply. They often connect advertisers not only to supply side platforms, but also to ad exchanges and ad networks. This can significantly increase the reach of ads. In 2007, MediaMath[4] introduced the first DSP that is still successful and serves advertisers such as eBay, Merkle, 1-800-Flowers and others.

Ad Networks. Ad networks exist to help advertisers achieve reach and scale in fragmented online ad marketplaces. No advertiser wants to run around piecing together inventory for a campaign from hundreds of individual websites or publishers. Ad networks help by bringing together supply, to enable advertisers to buy online ad impressions faster, more efficiently and more cheaply. Most importantly ad networks build targetable *audiences*, or audience segments based upon users that have similar profiles. They often offer an intermediary service to supply and demand side platforms, as well as dealing with publishers and advertisers directly. Sometimes, supply is provided by a single, however huge, publisher. The Display Network is a collection of websites, including specific Google websites like Google Finance, Gmail, Blogger, and YouTube, that show AdWords ads. This is Google's ad network, which also includes mobile sites and

[2] `http://rubiconproject.com/seller-cloud/`

[3] `http://blogs.wsj.com/cmo/2015/03/18/pg-cut-traditional-ad-spending-by-14-in-2014/`

[4] `http://www.mediamath.com/`

apps. An ad network can be focused on specific media types, such as video or mobile. InMobi[5] is a performance-based mobile ad network.

Ad Exchanges. The ad exchange could potentially be compared to a stock exchange. While it's not exactly similar, it does serve as a platform to increase the efficiency of the online ad market by making it easier for advertisers to find the audiences and impressions they need at the right price. Ad exchanges create liquidity, the moment when there are enough impressions, buyers and sellers in a marketplace to ensure that it runs on a pure demand-supply basis. Publishers send their ad impressions into the exchange hoping that some advertiser will buy them. Buyers come to the exchange to pick which impressions they wish to purchase using technologies like demand-side platforms. Those decisions are often made in real-time based on information such as the previous behavior of the user an ad is being served to, the time of day, device type, ad position and more. Ad exchanges are increasingly becoming a default link between supply and demand side.

There is a number of ad exchanges running. For instance, Double click is an exchange owned by Google that sells billions of impressions daily. Facebook recently launched its own ad exchange, FBX.

Retargeting Agencies. A standard retargeting ad campaign is based on the past actions of the user. The user has come to an online store, and looked at a particular pair of shoes for 15 minutes. She even added the pair to the shopping cart, but then for some reason left the webpage without finishing the purchase. All that publisher has is the cookie of the user. Later, the publisher can decide to turn to a retargeting agency and request to show a particular ad (perhaps, depicting the item left in the cart) to the user. The retargeting agency will take the cookie of the user and will attempt to track her down, for instance by bidding on ad exchanges. Companies like Criteo and AdRoll made retargeting their primary line of business.

Recently, companies with a novel approach to retargeting have appeared. For instance, search advertising is successful because the user declares the topic of interest via

[5] `http://www.inmobi.com/`

the query. However, the interest of the user is only accessible to the search engine, and not other players of the market. Chango, Rubicon Project[6] aggregates user search query information from tracking pixels. Later it sells audiences to advertisers. For instance, Chango is able to detect a user who was searching for reviews for a particular TV set, and later Chango helps BestBuy to show the user the ad of the same TV.

Differences Between DSP's, SSP's, Ad Networks and Ad Exchanges. DSP's, SSP's, ad networks and ad exchanges have many features in common. As the market develops the line between these entities often gets blurred. For instance, a single company often runs both DSP and SSP. For instance, Rubicon Project has SellerCloud, which is a SSP and BuyerCloud, which is a DSP. Furthermore, DSP's incorporate much of ad network's functionality, including access to a wide range of inventory and targeting capabilities. But DSP's serve ads and optimize for ads, while ad networks optimize for their marketplace.

Ad exchanges are different from ad networks, as ad networks typically aggregate inventory from a range of publishers, mark it up and sell it on for a profit. Ad exchanges are supposedly more transparent than networks because they allow a buyer to see exactly what price impressions are being sold for.

Data Brokers. Data brokers are companies that collect and aggregate consumer information from a wide range of sources to create detailed profiles of individuals. They are packaging and selling user information as a commodity. The information is sold to publishers, advertisers and even government. Companies like Acxiom collect browsing data, transaction data and the like, generally via terms and conditions accepted by users with Acxiom third-party partners. That data is then stored and sold to companies.

Generally, data brokers are companies that individuals do not interact with or do business with directly. They acquire their information from brick and mortar stores and websites, as well as from local governments. Sometimes they perform active studies, however most of the information is collected online by tracking users. The user activity on the web is frequently tracked via *tracking pixels* and *like buttons*. On mobile devices

[6] `https://www.chango.com/`

user information is collected by apps. Tracking pixels are tiny, invisible-to-the-bare-eye images. The pixel is loaded when the user is visiting the website with the pixel. It communicates with cookies on a device and pulls the information from those cookies. This information is later aggregated. “Like," “Share" and “Pin" buttons can be seen on almost every single page on the web. They operate much like tracking pixels and allow companies, such as Facebook, to track their users when they are visiting other websites on the Internet. This information is later aggregated and used to personalize content and ads.

On mobiles, most information is collected by applications. For instance, if a user downloads and installs the Facebook messenger app it will ask the user for permission to access the phone’s call log, contacts stored on it and personal information stored on the device.

Publishers collect their users’ information by placing cookies onto users’ machines. Cookies generally do not contain any information that would identify a person. Usually they contain a string of text or a “unique identifier" that acts like a label. The publisher is able to recognize the user across multiple sessions and aggregate the information. This information is sometimes sold to data brokers. Both advertisers and publishers purchase information from data brokers to enhance their knowledge of users.

Measurement and Analytics. Both publishers and advertisers are trying to understand their users better. Publishers are interested in learning what interests their users, whether they have challenges using their products, etc. Advertisers are using measurement and analytics companies to measure or verify performance of their ad campaigns. ComScore and Nielsen are the primary trusted planning sources for advertisers and agencies spending ad dollars.

With the advances of technology more and more sophisticated tools are developed. Methods are based on placing JavaScript or other code snippets onto the pages, tracking pixels and cookies.

The primary distinction between analytics companies and data brokers is that data brokers collect the data for themselves to later sell as a commodity, while analytics companies provide information collection as a service, and the data stays with the company paying for the service.

2.2 Ad Markets

Online advertising became an integral part of today's Internet. Now ads accompany search results returned by search engines, digital newspapers hurry to show the trailer of the new movie, an online store has a recommendation in the form of an ad, and a social network has a "sponsored" job listing. Ads take many shapes and forms, and are targeted at different aspects of a user's life and actions. The world of online advertising naturally separated into different, sometimes overlapping markets: search ads, display ads, video ads, mobile ads, native ads, etc. Each market operates in its own unique way, and has unique dynamics and properties. Here we discuss major markets.

Search Engine Marketing

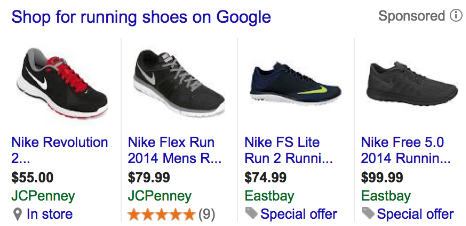
Search is inseparable from surfing the web. Google alone processes on average of over 40,000 search queries every second[7]. This translates to over 3.5 billion searches per day and 1.2 trillion searches per year worldwide. Most search results are accompanied by *search ads*. Search ads, the most common form of PPC (pay per click) advertising, are targeted at the query. The users that click on search ads are high-quality leads, because they were actually searching for the subject matter. Most search ads are text only (see Figure 2.4a.), and are very restrictive in terms of style and character count. However, they are also non-annoying and do not avert the user.

At a higher level the market works as follows. Advertisers create ad campaigns. Each ad campaign has the number of parameters, such as bid, budget, target audience and set of *keywords*. Search engines develop a machinery to be able to match a variety of user queries to keywords, model quality and relevance of ads via click-through rates (often referred as CTRs), *etc.*

When a user query arrives it is matched to related keywords. Next, the set of

[7] `http://www.internetlivestats.com/google-search-statistics/`

(a) Text ad



(b) Product ad

Figure 2.4: Example of *text* and *product* search ads shown in response to search query “running shoes" on Google

eligible ads is retrieved: the ads have to be active, have money on their account, but most importantly the query has to match some of ad’s keywords and the user, posting the query, has to belong to the target audience. To choose and price the ads to show to the user the search engine runs a real time auction, usually a *GSP* auction [14].

Finally, search engine produces the response that consists of two parts: *organic* and *sponsored*. Both results are targeted at the user’s query. The organic results are links to websites that the search engine finds to be most relevant to the user query. The sponsored results are ads that were chosen for the query.

Product ads are a new category of search ads. They visually showcase products in a larger format with images, text, pricing and the company name (See Figure 2.4b).

Display Advertising

Users spend a significant amount of their time online, visiting various websites, reading news, checking the weather, planning vacations, etc. Display advertising is a form of advertising that can be seen on websites across the Internet and it offers a unique combination of reach and targeting. It has extensive reach because it enables advertisers to serve ads to a select audience no matter where that audience travels on the web. And it offers precise targeting because of the availability of data on the Internet users. There are many formats display advertising can take, for instance text, image, flash, video or audio. Similarly to search ads, display ads are chosen for a particular invocation of the website. However, display ads can be targeted not only to user actions on the page at the given visit, but also to the user herself (see Figure 2.5b), the context of the page

(a) *Context* targeted display ad (banner format)

(b) *User profile* targeted display ad (banner format)

Figure 2.5: Example of *context* and *user* targeted display ads. The example is taken from The New York Times. On both occasions the website of The New York Times was visited by a user interested in *technology*.

(see Figure 2.5a), etc. As seen from Figure 2.5 a single website can use a mixture of targeting strategies. The exact sources that are used to fetch ads differ from website to website, and from impression to impression (see Section 2.1).

Social Advertising

Online social networks have become a global phenomenon with enormous social as well as economic impact in the past decade. In the second quarter of 2015, Facebook alone had 1.49 billion monthly active users [8]. Most social networks contain sponsored content or show ads. According to Nielsen's report[9] ninety-two percent of consumers around the world say they trust earned media, such as word-of-mouth and recommendations from friends and family, above all other forms of advertising. In addition to that, social platforms like Facebook, Twitter and others provide new signals for ad targeting, such as one's friends and followers, topics of interest, events in their lives, likes, and so on.

Social advertising platforms operate on top of an existing social platform. The social advertising platform allows advertisers to reach their target in the following ways:

[8] `http://www.statista.com/statistics/264810/number-of-monthly-active-facebook-users-worldwide/`

[9] `http://www.nielsen.com/us/en/press-room/2012/nielsen-global-consumers-trust-in-earned-advertising-grows.html`

1. by promoting advertisers' content; or

2. by allowing advertisers to recruit participants of the social network to place the advertisement.

Further, the platform usually allows advertisers to collect information about users that responded to the ad campaign, helps in choosing targeting strategies and delivering advertising messages. For instance, platforms like Facebook and Twitter allow the targeting of users based on their profiles and relationships presented on these same platforms. Platforms also allow advertisers to promote their content.

Recently a new type of intermediaries that use the market approach to word of mouth has emerged. Let us refer to them as *social markets*. Social markets let any user on a social network be a "publisher" of advertisements and provide a matching market so advertisers can find social publishers to advertise their products. They often let the social publishers modify the ad. As a result, a single base ad may be morphed into many, each modified by the human to better address the audience. Finally, platforms provide a mechanism for advertisers to pay the publishers based on engagement of the social users with the ads. Further, the social post can be engaged with by the users. For instance, users can re-share it, re-tweet it, comment on it, *etc.* For instance, the MyLikes platform allows advertisers to recruit Twitter and YouTube users to publish sponsored content.

Video Advertising

Video advertising is an advertising that appears before, during or after digital video content in a video player (*i.e.,* pre-roll, mid-roll and post-roll video ads). Video Overlays are small ads that appear on top of digital video content. They are also categorized as Digital Video Advertising. They can be display, video, rich media, text or another ad format but are contained within the video player. Display-related ads on a page that are not in a player that contain video are categorized as rich media ads.

Video advertising is a huge market. According to eMarketer estimates, in 2014 the US digital video ad spending was $5.81 billions. In the US, spending on video advertising in 2015 will grow around 25% to reach $7.7 billions.

Mobile Advertising

Mobile advertising is an advertising tailored to and delivered through wireless mobile devices such as mobile phones and tablets. Mobile advertising is generally carried out via text messages or applications and typically takes form of static or rich media display ads, text ads, search ads or video ads. Mobile is growing faster than all other digital advertising formats in the US, as advertisers begin allocating dollars to catch the eyes of a growing class of "mobile-first" users. Mobile advertising is also a prime monetization strategy for many mobile applications. For instance, RunKeeper[10], an app that allows a user to track her running activities, displays offers from brands to its users as they reach certain milestones in their fitness regimens.

2.3 Goods Sold on The Market

Advertising is a medium that is hard to measure. In particular, it is hard to measure the impact of a single impression of an ad to a user. Many factors can affect the efficiency of the ad: from type of the day the user is having, to the number of times she was exposed to the ad, to the opinion of the colleague she has heard over the lunch. However, the *performance* of digital advertising is easy to track, given the current technology. Of course, tracking of performance does not solve problems such as *attribution* — finding a sequence of ad impressions that have contributed to the click or conversion events. Typically *last-click attribution* is used, it attributes the event to the last impression of the ad.

Digital advertising allows the marketer to measure and track the following events, that also form the *funnel* of user engagement:

- **impression** is when the ad is fetched from its source, is countable and can be seen by the user. User does not have to engage with an ad.

[10] `http://www.mobilemarketer.com/cms/news/advertising/18527.html`

- **click** is when the ad get is clicked by the user. Clicks can be honest and can be the result of the user interest. However, click fraud and accidental clicks are a significant phenomenon.

- **conversion** happens when the user performs the action advertised. Depending on the type and business goals of the ad the conversion can have very different semantic meanings. For instance, conversion can be anything from a "registration in the system" to "purchase of Ferrari."

Some ad types have additional measurable events. For instance, for video ads advertisers can track the number of users who have completed the view of the ad, and count the number of users who have completed viewing some particular part of the ad. In addition they can track whether video was on mute, played in full-screen mode, etc.

2.3.1 Pricing Models

At the rise of online advertising there were two standard pricing options: CPM (cost per 1000 impressions) and CPC (cost per click). CPM was and is a standard for display advertising, and CPC was and is a standard for search ads.

For the publisher, every pricing model is translated to $eCPM$ — effective revenue per (1000) impression(s). However, pricing schemes have strong implications to advertisers and market. They affect budgeting and targeting strategies of advertisers, allocation mechanisms used, etc. For brand awareness campaigns advertisers are not expecting any actions from the user and are eager to pay CPM prices. On most of the other occasions, advertisers want an assurance that they are getting quality impressions. Clicks on ads are a good proxy. The increasing focus is on performance-based advertising. There is an increasing interest in CPA (cost-per-action) advertising. Other popular pricing models are cost per lead, cost per engagement, cost per completed view.

For simplicity, assume that publisher's only goal is to maximize short-term revenue. Then, given a single impression opportunity it would allocate it to the advertiser, who pays most revenue in *expectation*. It is frequently the case that a publisher (or ad network) has to choose between ads with different pricing models. For instance, j_1 is paying per conversion and j_2 is paying per click.[11] Intuitively, the publisher can translate CPC, CPA, and other types of bids to an $eCPM$ bid by predicting the click-through rates (CTR) and conversion rates (CVR) for ads shown in the particular ad spot, *e.g.,*

$$\begin{aligned} eCPM &= CPC \times CTR \times 1000 \\ &= CPA \times CVR \times CTR \times 1000 \end{aligned}$$

Now the publisher can rank ads in decreasing order of $eCPM$, and pick the ad on top of the ranking. Note that this $eCPM$ calculation depends on the accuracy of the CTR and CVR predictions. The less accurate the prediction, the more risk the publisher will take on. Intuitively, it is more difficult to predict less frequent events: the further down the funnel, the less frequently they happen, the harder it is to predict them and the more they are influenced by various external factors.

2.4 Research Directions

Depending on mechanisms and participants, each market presents us with a reach space of research problems. Roughly, research problems can be split into three categories:

- *Strategy Space.* This set of problems consider behavior and response of advertisers to the market. For instance, [14, 48] discuss strategies of advertisers in search ad markets that are implementing Generalized Second Price auction (or GSP) as a mechanism to allocate incoming impressions to ads.

- *Learning and Estimation.* These problems mostly deal with parameter estimation. In a GSP auction the ads are ranked in decreased order of expected revenue. The payments are performed per user click, hence estimation of the probability of the click on the ad for a given impression is a very important problem to solve.

- *Auction/Allocation/Optimization Problems.* Behavior and decisions of publishers and advertisers are highly dependent on the way the platform allocates incoming impressions to ads. For instance, if there are budget-constrained advertisers on

[11] http://www2009.org/proceedings/pdf/p221.pdf

the market (*i.e.,* advertisers that, if allowed to participate in all the auctions for which they are eligible, will run out of their daily budgets early and will not be able to participate in auctions later that day) the platform can decide to do nothing, but can also decide to optimize on behalf of advertisers. This aspect for search advertising market is studied in [28].

Markets such as search and display have been around for a decade and have received a larger attention from the research community. There are numerous research works exploring these markets, starting with [14, 48]. However new markets emerge and introduce new directions and challenges. For instance, consider research problems that arise in social markets.

Research Directions of Social Markets

We discuss market approach to social matkets in more detail in [32]. Here we consider research directions.

1. Strategy Space. Consider an advertiser j who has a true underlying value v per user engagement, true budget B^* and true target audience $G(j)^*$. Advertiser has many strategies available to them, reporting a bid b, budget B and target $G(j)$, all possibly different from the true quantities. The advertiser may benefit by misreporting these parameters. For example:

- Say j misreports $G(j)$ to be $G'(j)$. Advertiser j may be able to reach a large number of users in $G(j)$ and some users in $G'(j)$ for the same budget because of the overlap of $G'(j)$ and $G(j)$ in the broadcast communication platform.

- Say j reports B lower than B^*. Publishers p may still publish the sponsored post and exhaust the budget soon. The post will still remain in the system and will get publicity.

- Say p chooses a campaign C to sponsor. Publisher p has numerous options to modify C and affect the engagement of users with C.

Hence, underlying mechanisms in social ad markets have to contend with such strategies.

2. Learning and Estimation. There are many parameters that need to be learned. This includes

- For p and $G(j)$ pair we need to estimate number of impressions, clicks, retweets and other engagements. p's need these to figure out which campaigns to engage with. The platform can use these estimates to create listings of campaigns for p's.
- For each p and campaign $C(j)$ we need to estimate the *rate* at which the above quantities are accumulated. Since user engagement happens potentially in the future, the platform needs to match ads to publishers based on their potential leftover budget.

These items have be to learned as statistical quantities without regard to the strategies of players, or as.

3. Auction/Allocation/Optimization Problems. Each major player in the market has their own goal in the market.

- The publisher has to determine the series of ads to promote and optimize both short- and long-term revenue and satisfaction of their followers.
- The advertiser has to consider v, B^* and $G^*(j)$'s and construct ad campaigns in order to maximize their profit or brand goals, while cognizant of the other advertisers on the market.
- The platform needs to select a small set of a's to show to p, rank them into an ordered list, determine a pricing for user engagement on each, and so on.

These types of problems have been studied extensively in sponsored search and display ads markets. Studying them in markets for social ads leads to new research problems.

The Publisher. Fix a publisher p. Assume there is a bipartite graph in which one set of nodes represents p's followers $f(p)$ and the other set of nodes represents potential campaigns C. Let $G(C)$ denote the target audience set of C, we use it to denote both the set of features of the users and the set of users with those features. Then, there is an edge $(f(p), C)$ if $f(p) \in G(C)$. We can also assume that for each edge $(f(p), C)$, p

knows the quality of engagement. This defines a weighted bipartite graph, similar to weighted bipartite graphs defined in sponsored search and display ads. However there is a number of differences. C's arrive online or can be assumed to be known *a priori*. In social ad markets campaigns C can be matched with multiple publishers p, since a single publisher may tweet more than once for any campaign. In addition we can model the arrival and persistence of the campaigns. A bigger departure is that p may modify the campaign ads strategically to tune the quality of edge $(f(p), C)$ in a joint way among groups of $f(p)$'s. Furthermore, p's have to solve a *production* problem. p's know the arrival estimates of C's, the tweet consumption habits of their $f(p)$'s and the graph above. At what rate should p tweet sponsored posts? How many not sponsored posts should p have between sponsored ones? How many times should p repeat a post for the same campaign? One can formulate a suitable objective function and solve the offline or online versions. In contrast, sponsored search and display ad markets typically study yield optimization which for a given inventory determines which ads to place or procure, and exogenizes the production of inventory [21, 7].

The Advertiser. Consider an advertiser j with true budget B^*, target audience $G^*(j)$, and true value v per engagement from any user from $G^*(j)$. j needs to set up campaigns $C_1, \ldots, C_k$ for some k, each with its budget B_i, bid b_i and target $G(C_i)$ such that the total budget is no more than B, and the engagement from $G^*(j)$ is optimized. This is similar to *budget optimization* in sponsored search and display ads markets [20]. At the high level, one commonly approaches this as a best response problem given summary of the impact of other campaigns on the market, which ultimately becomes knapsack problems given estimates of performance of each campaign C_i. However, in social ad markets one wishes to optimize not the total impressions or clicks for $\sum_i |G(C_i)|$, but rather their union, $|\sum_i G(C_i)|$. Further, there is a possibility of *negative* audience. Each campaign C_i with target $G(C_i)$ reach an audience $T(C_i)$ that overlaps with $G(C_i)$ but also contains others and generates engagement from an unintended audience. It is reasonable to define a negative audience $N(C_i) \in T(C_i)$ as $N(C_i) \cap G(C_i) = \phi$. j needs to optimize over $G(C_i)$'s and penalize for $N(C_i)$'s. This leads to potentially hard variants of broad matching problems that was introduced in [17].

The Market. The market has a bipartite graph as well. Each publisher p is a node in the graph, as is each ad campaign C. For each edge (p, C), the market knows the engagement numbers for target $G(C)$ among the followers $f(p)$ of p. The market has to determine which campaigns to present to the p's with both short- and long-term revenues in mind. This is an ad allocation problem. There is a literature on ad allocation problems as they arise in sponsored search and display ads, and their formulation as online matching problems with possibly sub-modular welfare functions [38, 40]. In social ad markets, both C's and p's arrive online, C's persist and p's may arrive several times, and this departs from the online arrival of perishable impressions in sponsored search and display ads markets. Further, one can choose to optimize total engagement of users in $f(p)$'s. This objective function induces set-union-based objectives. Also, each occurrence of p is related to other occurrences, and this induces graph constraints on one of the partitions. Most crucially, ad allocation has to be solved in the presence of strategies of p's. This does not have an immediate analogy in ad allocation problems in sponsored search and display ads where users (impressions) are not thought to be strategic.

One approach to ad allocation is repeated auctioning with some budget admission control on top. Then the question is what auctions are suitable. It is natural to assume that when a publisher arrives, all applicable campaigns can be considered and some version of GSP auction may be run, as in sponsored search [14, 48]. However, in social ad markets p chooses a campaign strategically, in contrast to a sponsored search where users click on ads based on their utility to them. This leads to new modeling and formulation of the auctioning problem.

Each of the above directions can be abstracted into concrete problems that are open in theory.

Chapter 3

Adscape: Advertising Landscape

As discussed in the previous chapter, the targeting of ads has evolved dramatically (Figure 2.3): from generic, Flash-based ads shown to every single visitor of the website, to ads that are highly targeted, personalized, customized and chosen for a particular user impression in real-time. The selection process of ads can be affected by the website visited, user browsing history, user properties, time of day and even weather outside. For instance, *search ads* are targeted at the query that the user presents to the search engine. Users are very likely to click on these ads. Search ads are generally effective because they are related to the current activity of the user, and not off putting, because they look very much like organic search results and do not distract the user. However, even search ads are not that simple. Search results as well as search ads are affected by user properties. Both are "personalized." For instance, a search for "pizza" will generate organic and advertising results both related to the origin of the user of the request.

When browsing on other properties (websites, mobile applications, *etc*) the user is also facing generic and personalized content. Many services are building *recommender systems* that choose the content that is likelier to engage the user. The content is often accompanied by ads: display (images), text or video formats. However, outside of search it is much harder to capture the intent of the user. For instance, it is hard to decide which ads to show to the user, who is checking the weather for the region where she is located. Therefore, depending on a particular case, ads could be targeted at (i.) context of the website (*i.e.,* contextual advertising); or (ii.) properties of the user, such as interests, demographics and geo location (*i.e.,* targeted advertising).

There is a large body of work that explores optimal bidding strategies for advertisers [42, 48, 14, 10], propeties, properties of auctions used to allocate a particular impressions [33, 15, 35], or optimal ad allocation algorithms for publishers [38] or intermediaries [7], such as ad networks and ad exchanges. (this list of references is not exhaustive, and only points out starting points.) However, there is little to no understanding of how an ad market appears to its end consumer, or online user. It is natural to try to get a better understanding of the following questions: Who is being targeted on the Internet and how? Do different users have similar experiences, in terms of ads? What properties determine the ad experience the user has online? Our research agenda is to broadly understand the user-targeting aspect of online display advertising.

3.1 Adscape: Advertising Landscape

Ad impressions have a very short lifespan. Impressions of a particular ad can be hard to reproduce. Each time the page loaded, a new ad can be shown, even for a fixed user. In addition to time, each ad impression can be characterized along three perspectives that are determined by different players on the online ad market:

1. **publisher** — website or app where the ad appears (or supply side);
2. **advertiser** — the entity that owns the product that is being advertised (or marketers, demand side);
3. **user** — the user who came to publisher's property and saw the ad of an advertiser.

Each of these players affects the ad shown in a specific way. In Section 2.1 we have discussed goals and strategies of publishers as well as advertisers. Here we focus our attention on the *user perspective.*

Definition of Adscape. Consider a user $u(t)$ accessing a publisher's property $w(t)$ at time t. Say the publisher of $w(t)$ shows a set of ads $a(t)$ to $u(t)$. There is some *allocation function*

$$f_w(t) : u(t) \to a(t)$$

The set of all $f_w(t)$'s over all w's and all users $u(t)$ at any time t will be the *Adscape* that is our focus. The functions $f_w(t)$'s may depend on:

- user's demographics, interests, location, etc;
- publisher's property w, its contents and context;
- time t, and the past, including user's past actions, w's past contents, while f may vary over time;
- the set of ads $a(\dot{})$'s, including mutual constraints that allow or disallow each other;
- mechanisms, incentives and market conditions that govern the behavior of advertisers, networks, exchanges and publishers.

The function $f_w(t)$ may ultimately be simple (all users see the same ads for a day) or sophisticated (each ad is personalized based on multiple criteria).

3.2 Observing an Adscape

3.2.1 General Approaches

The starting point for understanding f_w is the ability to collect data from a number of publishers and users with broad interests. There are many such approaches.

Collection of Data from Real Users. One could consider installing recording software on the users' side, or directing user-request traffic to a proxy that is setup as data collection point. Due to privacy concerns, recruiting test subjects and developing recording software are challenges that make this approach difficult to implement and scale up.

Collection of Data from Services. The data can be acquired directly from publishers. But we require a number of attributes for each ad impression: content, user profile at the time the ad was served and ad information. Often publishers use third parties to fill impressions, and are not aware of particulars: which ad was served or what information ad network has for the user.

Collection of Data from Ad Networks. Ad networks, such as Google's AdSense, serve many publishers. However, we would need to know the state of user profile p

at a time t when ad was served. Ad networks are unable to provide such information. Moreover, advertising is a primary business and companies are unlikely to reveal details.

Crawling. One can consider crawling publishers' websites with synthetic profiles (*e.g.,* [11, 8]). This method is particularly attractive since one would have control over the publisher's targets, access frequency and user profile choices. However, how would one create synthetic profiles? We identify two possible approaches to profile creation:

A. **Through Browsing.** Synthetic user profiles can be established through targeted browsing similar to what is reported in [8], and use a popular user tracking service (*e.g.,* Google) as a proxy to find out the result user profile interests. Later one can use the created profiles for data harvesting and analysis. On the plus side, this method allows many different entities involved in ad serving process to record and track the synthetic profile. However, it is hard to have a fine control over the outcome. For instance, if one wants to create a profile associated with "exotic pets," what content should one consume? Which websites should she visit during the profile construction phase, in particular when a single website can belong to multiple categories?

B. **Through Manual Setup.** Some companies that profile users offer an interface for users to monitor or modify their profiles. For example, one can change demographic information, or add or remove individual interests from Google at Ads Settings Manager[1], or from eXelate at interests settings page[2]. Potentially, one could visit some or all of these interfaces and set up suitable profiles. Later one can perform data collection, using established profiles. While not all companies have such functionality, in some cases this method may be sufficient for the task.

3.2.2 Challenges

Let's consider ad crawling compared to web crawling. In general web crawling the goal is to identify and possibly index the *contents* (ads aside) of all the webpages w

[1] `https://www.google.com/settings/ads`

[2] `http://exelate.com/privacy/opt-in-opt-out/`

that are online. In what follows, we compare that with the problem of crawling and understanding $f_w(t)$'s and ultimately the Adscape.

Content vs. Ads. Web crawling is challenging because the existence of a page w may be unknown. Our problem of crawling for $f_w(t)$'s has similar challenges in discovering w's. However, there are additional challenges. For example, given some page w, it is nontrivial even to identify which elements are ads, and which are regular content. Further, given an element that *is* an ad, it is typically not represented in HTML code, but in some other form (usually JavaScript), and several executions may be needed to actually retrieve the ad.

Dynamic Content and Re-crawl Rate. Web crawling is challenging because w may have dynamic content that varies over time. To address this, robust web-crawling methods have been developed, including the ability to identify content change frequency and using that to specify re-crawling rates. In contrast, ads are far more dynamic — a popular web page may show many different ads depending on a user's geographic location the time of day, *etc.*, even if the content on the page does not change. This complicates calibration of ad rec-rawling rates.

Personalization. Arguably some web content is personalized to the viewer, thus web crawlers have to mimic viewers to collect this content. In many cases, the personalized content may not even be relevant to crawl. In contrast, ads are critically targeted to users' profiles, and it is imperative to mimic multiple user profiles in order to understand the Adscape.

Contamination. Ad crawlers that visit pages end up contributing to data observers on the web who build profiles of users based on the browsing behavior. Therefore, even visiting a page modifies the profile associated with the crawling session, which can contaminate the profile that a crawler adopts. Further, intermediaries like the ad networks observe browser sessions and adopt their strategies, leading to potentially further contamination.

Economics. Web crawlers should be calibrated not to overload the sites they crawl. Similar consideration holds for ad crawlers, but there is a deeper concern. Display ads are charged *per impression.* Thus, each time the crawler accesses a page and associated ads, advertisers incur a cost. Therefore, an ad crawler must be configured to limit costs for the advertisers, which in turn limits the sample of the space that is observed.

Ultimately, an ad crawler needs to (i) search over far more states than a corresponding web crawler, proportional not only to the number of pages, but also the number of user profiles, geo locations, day parts, etc. (ii) minimally distort the statistics of ads being displayed (and hence have minimal cost to the advertisers) (iii) and prevent contamination of a crawling profile. According to WorldWideWebSize[3], as of March 12, 2015, the Indexed Web contains at least 47.3 billion pages. The state an ad crawler has to explore is a multiple of this number. If we assume 1000's of geos, 10's of daily segments, 1000's of profiles, this altogether yields a multiplier in the range of 10^7 or more! Ultimately one must make some simplifying assumptions to shrink this state space.

3.3 Case Studies of Display and Video ads

In this section we present two case studies. One study is about the display ads market, and the second study is about the video ads market. We start with introducing the framework that we applied to both of the studies. We proceed by describing each study separately, highlighting the differences of two markets. We summarize by considering new research directions.

3.3.1 Our Approach

Let $\mathcal{W} = (w_1, w_2, \dots)$ be set of all websites, and $\mathcal{P} = (p_1, p_2, \dots)$ be a set of all personas. In general, we will not be able to use all pairs formed from $\mathcal{W}$ and $\mathcal{P}$ for crawling because of the imposed load on our systems, as well as the ad ecosystem. Hence we approach it in four steps:

1. Given a single pair (w, p) we crawl it — crawl site w with browser depicting persona p – several times in a row, study the distribution of ads over time, and

[3] `http://www.worldwidewebsize.com/`

propose a pattern of crawls (*crawling strategy*) we will ultimately deploy for the pair.

2. Create a large pool of websites $W \subset \mathcal{W}$ and pool of personas $P \subset \mathcal{P}$ which will be the basis for our research.

3. Crawl all possible pairs formed from W and P for a short period of time. Analyze the data to identify a small *focus set*: a subset of pairs that we will crawl operationally. The choice is done with a budget in terms of the number of crawls we can do with the crawling strategy above.

4. We crawl the focus set of pairs as per the strategy, log the crawls and collect data about the ads observed.

3.3.2 Display Ads Adscape

The Setup

In this section we focus on the impact of user interest-based ad personalization or *user interest-based Adscape*. More formally, we restrict $f_w(t) : u(t) \to a(t)$ as follows:

(1) we fix geolocation (by performing all data collection from a single location);

(2) we do not consider time-of-day effects.

Both dimensions are important to consider, and could be subjects of the future work. We focus on $f_w(t) : p(u(t)) \to a(t)$ where $p(u(t))$ is the *profile* (or *persona* which we use interchangeably) of the user. We will make $p()$ more precise in the future, but it encompasses users' interests. We refer to (w, p) as a *pair* where w is a website and p is a persona. While distribution of different types of personas in the Internet can be arbitrary and can be a parameter of $f_w(t)$, here we restrict our attention to targeting algorithms given a (w, p) pair. In the future, one could expand the study by exploring actual frequencies of different user types and shape of the traffic (*e.g.,* using comScore).

To form the study we follow the framework described in Section 3.3.1.

1. **Crawling Strategy.** Intuitively, we want to maximize total number of distinct ads. Our expectation is that the large corpus of distinct ads will allow us to detect targeting patterns. We are not aware of any previous work that discusses this problem.

   Ads shown to a pair (w, p) can be observed in many different ways. For instance, we can visit w and collect ads once every hour for a week, or we can visit w many times in rapid succession. Results will likely be different. For instance, if the user makes too many visits to a single page, she can be classified as a "bot," and as a result might only see the limited selection of ads. Alternatively, if advertisers use frequency caps and the cap is small, then sequential visits of the website with p can yield many distinct ads. If frequency capping is used rarely, then crawling may collect only a small selection of ads. Hence, to maximize the number of distinct ads, we will pursue 2 strategies: (1) short and (2) long. The strategies are characterized by two numbers: α — number of rapid sequential visits, and β — number of repetitions. Note, that initial information about p before each repetition is identical.

2. **Website Pool W.** The number of distinct webpages on the Internet is huge, and grows each day. However, in practice only a few of these are frequently visited. For this study, we create a pool W from top popular websites using Alexa[4].

3. **Persona Pool P.** We model a persona as a set of user profile interests that are associated with the user (or her browser, to be more precise). We build P from Google's advertising interests tree. For this study, we choose interest categories such that P is diverse and represents interests that are popular on the Internet.

4. **Selecting a Focus Set.** Let $S = \{(w_i, p_j)\}$ be the set of all possible pairs, such that $w_i \in W$ and $p_j \in P$. We want to select focus set $C \subseteq S$, such that the size of C is at most B, where B is the budget in number of pairs. B takes into account crawling strategies, as well as network, server and bandwidth constraints.

[4] `http://www.alexa.com/topsites`

To maximize the total number of distinct ads, we observe S for a limited time and choose C of size $\leq B$ that produces a large number of distinct ads.

It is easy to see that this is a budgeted maximum cover problem. Let $\{a_{ij}\}$ be a set of ads displayed on website w_i for profile p_j. Given a set $A = \cup_{(i,j)}\{a_{ij}\}$, we want to choose subset $C \subset S$, s.t., $|C| \leq B$ and $\forall T \subset S$, s.t., $|T| \leq B$, the following holds: $|\cup_{(w_i,p_j)\in C}\{a_{ij}\}| \geq |\cup_{(w_i,p_j)\in T}\{a_{ij}\}|$. This problem is NP-hard, hence we proceed with an approximate greedy solution that is known to be an $1 - \frac{1}{e}$ approximation to the optimal. The description of the greedy algorithm can be found in Algorithm 1. We start with empty cover *cover*. We sort pairs in decreasing order of number of distinct ads (o); ties are broken arbitrarily. While the budget is not exhausted, take the first element of the ordering h, remove it from the list, and add to *cover*. Process remaining pairs in o by removing ads u that occurred for pair h. Keep adding element to *cover* until the budget is exhausted or ordering o is empty. Output *cover*.

Algorithm 1 Select Subset of Pairs

procedure FINDMAXCOVER($\{w_i, p_j, a_{ij}\}, B$)
  $cover \leftarrow [\,]$, $u \leftarrow [\,]$
  $o \leftarrow pairsInDecrOrderOfNumOfAds(\{w_i, p_j, a_{ij}\})$
  for $i \in \{1, \ldots, B\}$ **do**
    $h \leftarrow o.pop()$
    $cover \leftarrow cover \cup h.getPair()$
    $u \leftarrow u \cup h.getAds()$
    $removeAds(o, u)$
    $o \leftarrow pairsInDecrOrderOfNumOfAds(o)$
  end for
  return *cover*
end procedure

Profile Creation

We utilize a crawler-based approach. We want to ensure that our synthetic personas are recognized across large number of web sites. We have implemented Profile Builder that for a given interest category does as follows:

(1) fetches the top 50 websites associated for this category (*e.g.,* using Adwords Ad

Planner);

(2) opens Firefox with an empty profile (*e.g.,* using Selenium WebDriver[5]) and visits fetched URLs;

(3) visits Google ads settings editor page and captures its content. This step is not required for profile generation, however it allows us to verify the interest categories assigned to the profile;

(4) zips the profile, including the cookies, and stores the profile for future use.

This approach has several limitations. For instance, profiles treated in this way do not take into account server side profiling or Flash cookies.

Harvesting and Identifying Ads

Ads are delivered to webpages by JavaScripts that are executed at the time of page load. We chose to stay agnostic to JavaScript execution, and load the pages in a browser. We have implemented a Firefox browser extension *Firefly* that is able to control the instance of Firefox. It can load w; once it is fully loaded, including all iFrames and JavaScripts, Firefly captures the source.

We have developed a three-step decision process that identifies ads automatically. In order to be classified as an "ad" a target visual element has to pass following tests:

1. *AdBlock Test.* Adblock's easy list [6] is a database of regular expressions that can be used to detect ads. We test an element's URL, iFrame URL, `div` class and a landing page URL against it to see if a match can be found;

2. *Dimension Test.* Display ads frequently have standard sizes (*e.g.,* a standard banner is 728×90), which enable them to fit into designs of many pages. We maintain a list of 25 different standard dimensions. The visual element has to match one of the entries of the list;

[5] `http://docs.seleniumhq.org/projects/webdriver/`

[6] `http://easylist.adblockplus.org`

3. *Self Ads.* A visual element cannot link to a page within the same domain. Ad elements have to have an external link.

Overall Solution

We have implemented a distributed ad harvesting and parsing infrastructure based on the methods described above. Ad harvesting is managed by the *Controller*. The Controller is configured with a *crawling plan* — a list of persona/site pairs to crawl with specified frequency — as an input, and executes it while balancing the load. The Controller manages the number of Firefox instances (also stated in the crawling plan) that are administered via our Firefly extensions. Most importantly, the Controller can open Firefox with profile p as per the crawling plan. The Controller sends commands to Fireflies to visit w's respecting profiles p that the Firefox browser is using. The Fireflies follow the order and report back harvested visual elements which are then stored into the database.

Parser instances function independently of the harvesters. They take unprocessed entries populated by harvesting, and parse them. In addition they (1) identify ads and (2) resolve ad landing pages. The Parser also downloads all visual elements and stores a local copy of them.

Results

Dataset. We used our crawler to collect data on the focus set. Data was collected over a two day period from 10/1/2013 to 10/3/2013. This data collection produced 875,209 impressions, and 175,495 distinct ads. We observed ads from 3,700 advertisers served using 106 distinct ad servers.

Importance of Profiles and Websites. We observed that *all* profiles from P and 180 our of 314 websites were selected into the focus set. To measure the importance of websites in data collection we rank all websites by the number of distinct ads collected from each of them across profiles. We do similarly for profiles. Figure 3.1 shows that the number of distinct ads collected grows almost linearly with the number of profiles.

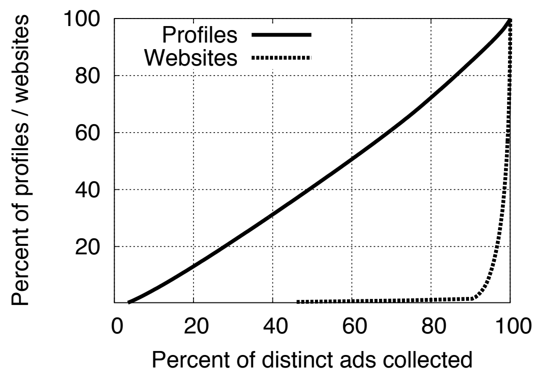


Figure 3.1: Impact of distinct websites and profiles to the number of distinct ads harvested.

Profiles \ Ads	Arts & Entertainment	Automotive	Business	Computers	Education	Financial Services and News	Games	Health	Home	Investing	News	Political	Restaurants	Shopping	Sports	Travel
Arts & Entertainment	0.097	0.051	0.137	0.154	0.052	0.067	0.014	0.061	0.031	0.007	0.005	0.003	0.006	0.218	0.024	0.053
Beauty & Fitness	0.021	0.027	0.126	0.141	0.027	0.080	0.002	0.122	0.051	0.067	0.004	0.000	0.004	0.155	0.000	0.160
Business & Industrial	0.091	0.028	0.200	0.106	0.075	0.065	0.010	0.032	0.014	0.011	0.052	0.002	0.023	0.188	0.009	0.062
Computers & Electronics	0.149	0.023	0.238	0.143	0.016	0.078	0.010	0.063	0.010	0.002	0.039	0.002	0.022	0.104	0.014	0.061
Finance	0.221	0.042	0.076	0.107	0.012	0.107	0.009	0.039	0.001	0.004	0.198	0.001	0.008	0.054	0.008	0.043
Food & Drink	0.180	0.021	0.067	0.160	0.014	0.017	0.002	0.014	0.004	0.036	0.188	0.000	0.007	0.110	0.087	0.057
Games	0.110	0.054	0.105	0.085	0.015	0.043	0.195	0.085	0.004	0.005	0.005	0.066	0.033	0.153	0.004	0.014
Health	0.026	0.016	0.198	0.155	0.023	0.101	0.004	0.140	0.009	0.006	0.006	0.005	0.023	0.125	0.011	0.131
Hobbies & Leisure	0.036	0.020	0.303	0.181	0.029	0.064	0.009	0.026	0.001	0.008	0.009	0.000	0.008	0.243	0.009	0.044
Home & Garden	0.091	0.013	0.136	0.167	0.022	0.041	0.022	0.053	0.027	0.002	0.020	0.000	0.026	0.237	0.091	0.019
Jobs & Education	0.203	0.008	0.073	0.111	0.045	0.092	0.005	0.147	0.003	0.003	0.083	0.000	0.006	0.090	0.002	0.068
Pets & Animals	0.013	0.040	0.325	0.074	0.015	0.029	0.022	0.028	0.150	0.002	0.014	0.000	0.017	0.227	0.002	0.020
Real Estate	0.192	0.016	0.078	0.326	0.018	0.013	0.001	0.023	0.013	0.001	0.106	0.001	0.006	0.144	0.006	0.015
Shopping	0.137	0.019	0.149	0.135	0.030	0.069	0.009	0.062	0.014	0.018	0.014	0.001	0.010	0.236	0.008	0.050
Sports	0.193	0.101	0.104	0.132	0.012	0.033	0.016	0.035	0.022	0.005	0.166	0.001	0.013	0.050	0.018	0.029
Travel	0.128	0.016	0.077	0.116	0.030	0.046	0.008	0.064	0.007	0.004	0.032	0.000	0.025	0.281	0.001	0.080
Empty	0.103	0.011	0.243	0.114	0.030	0.024	0.008	0.028	0.009	0.013	0.051	0.002	0.158	0.103	0.031	0.043

Figure 3.2: Correlation between use profile interests and ad categories. The heat map shows per profile interest the distribution of impressions over different ad categories. It is row-wise normalized based on number of ad impressions.

However, for websites we observe the opposite effect: 2% of all the websites produced 90% of all distinct ads observed! This means that many websites are redundant from the perspective of distinct ad collection, and one can significantly decrease the crawling space by carefully choosing w's.

Targeted Ads. Intuitively, a targeted ad is shown to some profiles more frequently than to others. For each ad a shown on a fixed website w, we count how many times it was shown to each of the profiles. This gives us empirical distribution $g_a(p)$ of ad a over profiles p. If ad a is not targeted based on users' profiles, it is fair to assume that the observed distribution g_a should be close to uniform. To compare g_a and uniform distribution we use Pearson's χ^2 test. We say that ad a is *targeted* if the resulting p-value is less than 0.05. Note that for some ads our sample size is too small, which makes it impossible to reject the hypothesis of uniform distribution, even if g_a is not actually drawn from uniform distribution. Therefore, we might get false negatives *i.e.,* ads that are targeted but we falsely classify them as non-targeted.

We calculate the fraction of targeted ads for each of the websites that are used in at least 10 pairs. We find that 50% of analyzed websites have at least 80% of their ad inventory targeted at profiles.

Profiles and Ad Categories. It is natural to expect that advertisers target different types of users. We are looking for the relation between interest categories of p's and categories c's of ads. For each ad impression, we consider the category c of the corresponding advertiser. We attribute the appearance of c to all the profiles that observed the ad. We normalize this by the total number of p's that saw the ad, to discount for the fact that many different profiles saw it, possibly without being the focus of targeting. We get distribution of impressions over different ad categories (Figure 3.2). Several characteristics are immediately evident: (1) Some p's and c's exhibit high correlation, for example *Games*, *Health* and *Shopping*; (2) Some profiles are targeted by related categories, for example *Beauty & Fitness* profile is highly targeted by *Shopping* and *Travel*, and *Pets & Animals* profile is highly targeted by *Home* related ads; (3) Ads from categories *Arts & Entertainment*, *Business*, *Computers* and *Shopping* are less targeted; (4) Interestingly, the empty profile is highly targeted by *Restaurants* ads, which are rarely seen by any of the other profiles.

Further details of our study can be found in [8].

Future Extensions

Distribution of Properties of Real Users. We restricted our attention to targeting algorithms given a (w, p) pair. But in reality, the distribution of users in the Internet is not uniform. For instance, there are more users interested in *Travel* than in *Exotic pets*. One could expand the study by exploring actual frequencies of different user types and shape of the traffic (*e.g.,* using comScore).

Capturing Retargeted Ads. Crawler-based approach to profile generation is unable to discover *retargeted* ads, since in order for a profile to get retargeted it has to visit the website of the advertiser first. One could expand the study by

- creating a capability to locate advertisers and publishers who are involved in retargeting advertising;
- collecting the data and studying the phenomenon.

Impact of Time. Due to the size of the search space, for our data collection we have excluded the time from consideration. However most ad networks allow advertisers to target users by day parts. Also, it is only logical to assume that user attention can be captured more successfully on the weekend, rather then at 8AM on a business day. What is an impact of the time and day to the adscape? Does adscape of Sunday differ from adscape of Monday? Can this difference be quantified?

3.3.3 Video Ads Adscape

The Setup

YouTube's leading position among video sites makes it a strong candidate to study and understand its advertising, but there are other reasons. In particular, unlike most publishers, the user's profile is accessible and modifiable by the user. In our study we choose *crawling with manual setup of profiles*, as discussed in Section 3.2.1. This allows creation, modification and easy tracking of profiles. Further, ads on YouTube are placed by Google through AdWords, hence Google profiles are also used by advertisers for ad targeting. Altogether, all operations over the profile happen within Google, and Google allows a user to have access to the profile.

Profile and Content Analysis. Crawling of videos presents the challenge of content classification. In addition, it is not clear how one would compare *profiles* to *videos*. YouTube provides the language of *interests and verticals*, that allows us to perform such an operation.

- *Verticals* describe content. Vertical is the term used by AdWords to refer to content categories. Using verticals, advertisers can target users based on the content they are watching.

- *Interests* describe the user's subject concerns (e.g., the user likes *fishing* and *country music*). Using interests, the advertiser can target relevant users, regardless of what they are currently watching.

The hierarchies of verticals and interests are almost identical. Verticals of the video and YouTube hosted ads can be harvested off the pages. This eliminates discrepancies from automated category assignments to videos and ads.

To form the study we follow the framework described in Section 3.3.1.

1. For our pool of profiles P, in order to cover a diversity of interests while maintaining a small number of profiles, we create profiles corresponding to first-level categories of Google's interest tree (25 in total). We add an *empty* profile that represents a new user that has not yet been assigned any interest categories. In total, $|P| = 26$.

2. For our pool of videos V, we focus on covering the space of verticals, rather than the space of videos. We use Google's Display Planner[7] to find the top 50 channels for each of 25 first-level categories. For each identified channel we take the 50 latest videos, resulting in 1250 videos total. We proceed by finding verticals for each of the videos in the list, and building a set cover on verticals. We form V consisting of 101 videos, s.t., (1) V contains maximal number of verticals; (2) most verticals are assigned to at least 3 videos in the cover, and only a few are assigned to fewer than 3.

3. For data collection we form all possible *pairs* (p, v), s.t., $p \in P$ and $v \in V$. We visit each pair 20 times, and stay on each video for 2 minutes, which is sufficient for pre-roll, companion and sponsored ads to load. We create a distinct instance of profile p for each visit. As a result we obtain 52,520 data points in one week from: $|P| \times |V| \times 20$. We refer to each data point as an *instance.*

[7] `https://adwords.google.com/da/DisplayPlanner/home`

Overall Solution

Our overall solution for video adscape is similar to the solution we had for display adscape. We have: 1. Profile Builder (that is now implemented using “manual setup"); 2. Ad Crawler, that is able to visit page of video v with a profile p; 3. Ad Parser, that is able to detect ads in data harvested by Ad Crawler; 4. Scheduler, that takes list of pairs as an input, and manages a number of machines to crawl according to the list.

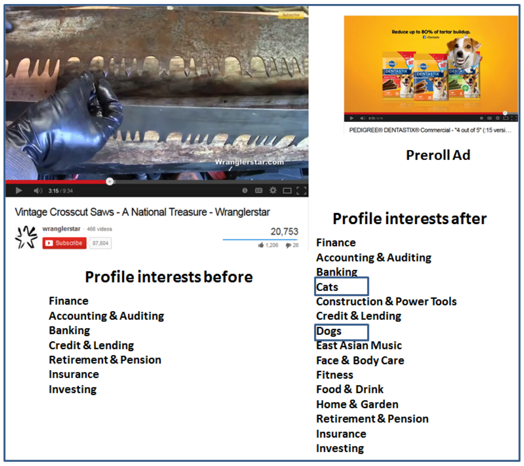
Capturing and Parsing Video Ads One of main challenges was to identify ads and capture ad-related meta information. Video ads, however, are not present on the page itself, but are streamed from the Flash player directly. Analysis of communication logs — a record of pairs of requests and responses of the web browser as the video page is loaded — revealed that the metadata for many of the ads is contained in XML files received by the browser during page loading, providing instructions to the video player. XML files can be recognized by name, for instance, sponsored videos can be recognized by a request starting with `http://www.youtube.com/pyv`. The format of the file differs slightly depending on the type of the ad: XML of preroll ad may contain a description of a companion ad, XML of *video wall* will contain channel information as well as attributes of 1-4 videos that are to be molded into a single creative.

The communication log is captured by Ad Crawler during the harvesting. For each ad detected, ad parser will parse the captured XML file, depending on the type of the ad and capture the following meta data of the ad: video URL, landing page, title, duration, view counts and description are recorded for pre-roll, companion and sponsored ads. For overlay text ad we are only able to get the text and the landing page.

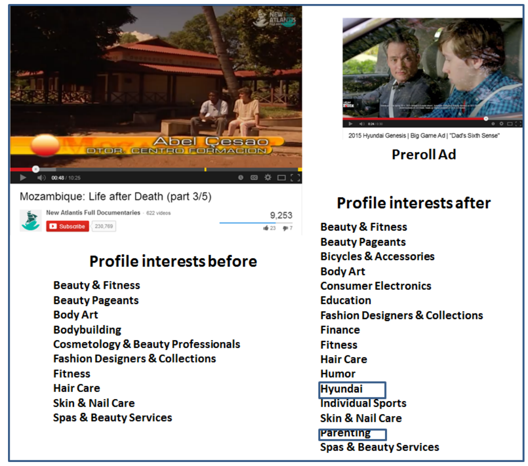
Results

Dataset. In our data collection, we have observed 38,758 ad impressions, 23,600 of which are prerolls. Interestingly, 61% of (p, v)'s observed only one ad in 20 visits, and about half of the ads were collected from only 7 videos. Furthermore, empty profiles collected more ads than most other profiles.

Video Ad Targeting. We study the relationships between video ads and content, and



(a) Video *"Vintage Crosscut Saw"* was watched by a profile with *Finance* interest.



(b) Video *"Mozambique: Life after Death"* was watched by a profile with *Beauty* interest.

Figure 3.3: Two examples of impact of a YouTube preroll ad to the profile

video ads and profiles. To do so, we compare verticals associated with the video viewed and the interest categories added to the profile after viewing. We perform the comparison using the first level categories of the interest tree. We consider overlaps between: (1) verticals of preroll ads and verticals of videos; and (2) verticals of preroll ads and profile interest categories (prior to video view). We observe that in approximately 12% of cases there is no overlap between two sets. In almost 70% of the cases ad and video have exactly one first-level category in common, in the rest of the cases video and ad have two categories in common. These results indicate a significant connection between the content of the video and the content of the ad (*i.e.*, presence of *contextual targeting*).

Interestingly, we found no instances where the interests of profile overlapped the verticals of ads. Thus, there is no indication that prerolls were targeting user profiles.

Dynamics of Profiles We have observed that video watched affects user profile significantly: a visit to a single video can add several — in some cases up to 25 — new interest categories to a user profile. However, the connection between video content and specific changes in profiles is not always evident. There is significant overlap between video and ad verticals. Furthermore, we also find that on many occasions user profile is affected by the *video ad* that was shown. For instance, consider example depicted in Figure 3.3.

3.4 Future Directions

As we have already discussed previously, we cannot harvest all of the ads, as the search space is huge, and is proportional not only to the number of websites, but also to the number of user interests, day parts, etc. In addition, adscape is highly dynamic. It is only natural to expect that all the players in the market adjust their strategies to improve efficiency of their performance in the market. Here, we map out potentially interesting directions that we have only started exploring.

Objective Function of Ad Collection. In our study of display ad market we have chosen to maximize the total number of distinct ads observed. We believed that the larger number of ads would allow us to better capture the adscape. However, that is not the only option. There are number of alternatives:

1. Observe maximum number of ads;
2. Observe maximum number of *distinct* ads;
3. Observe maximum number of distinct ads *for a particular persona*;
4. Observe maximum number of distinct ads *shown on a particular website*;
5. Find a set of websites on which a particular ad is shown;

Depending on the choice of the objective function, one is likely to end up with different focus sets and distinct crawling strategies. We believe that the choice of the objective function is crucial. The choice and its impact have to be investigated in a separate detailed study.

Strategy Observing a Single (w, p) **Pair** As discussed in Section 3.2.1, the strategy can have multiple parameters: given a pair (w, p) we start at time T, and perform β sets, each set consisting of α repetitions. We wait for S milliseconds between the sets, and s milliseconds between repetitions in the set. Note, that initial information about p before each repetition is identical. For instance: perform $\beta = 100$ sets, each with $\alpha = 10$ reps. We start at $T =$ 00:00 and harvest pair every hour ($S = 60$ *min*), with $s = 2$ *sec* waits between crawls.

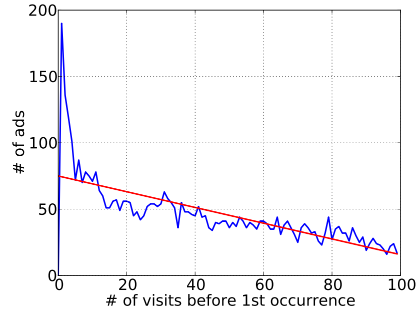


Figure 3.4: Average arrival rate of unobserved ads as a website gets visited 100 times.

Consider the following experiment. We chose 100 random pairs formed using W and P and visited each sequentially 100 times. We observed $\approx 3K$ unique ads. We studied the average arrival rate of new ads (see Figure 3.4). We observed that in the beginning of the session a given profile was shown many new ads. At approximately visit 5 the arrival rate of the new ads drops drastically. We also found that beyond the tenth visit, the rate at which new ads were served slowly decreases. This roughly follows a linear function $y = -0.6x + 75$ (red line on the plot). If our objective is to maximize the number of distinct ads given the budget B on the total number of pairs visited, then clearly *short* strategy $(\alpha, \beta) = (10, 5)$ would be a good chose.

Further, crawling strategy depends on the objective function of the ad collection. Can we define a formal approach to finding of crawling functions?

Choosing Focus Set. In our study of display ads we have reduced the choice of the focus set to the set cover problem. Is it a general approach to finding the focus set?

Adscape Properties. Social graphs are known to have a number of properties: number of triangles, diameter, and degree distribution are believed to be important properties of the social graphs. What would be important properties of adscape? For instance, is the average number of profiles targeted by an ad important?

Adscape Generative Model. It is resource consuming to create an adscape from scratch each time. However, one could model underlying processes (such as launch of new new campaigns, *etc.*) and create an incremental, or *dynamic version* of the adscape. More formally, given the state of the adscape at time t, can we calculate the state of the adscape as time t +1? Estimation of such processes is complex and may be impossible in practice. However, there may be an *iterative* approach to data collection that would allow one to minimize the effort of obtaining adscape at time $t+1$, given the adscape at time t.

Adscape Evaluation. As we have mentioned before, the search space is so large that is seems impossible to explore it all. Hence, one will end up making choices and limiting space: fixing geo location, time, limiting $\mathcal{W}$ to a subset of possible URLs, or limiting $\mathcal{P}$ to a particular subset, or omitting not frequent personas altogether. One needs a methodology that would help to estimate the impact of these limitations.

One could also use evaluation tools to validate models of an adscape.

3.5 Related Work

Web crawling. Our ad crawling capability is most directly related to standard *web crawling*, which is widely used to gather content for search engines and a host of other applications. Early web crawlers emerged nearly two decades ago including WebCrawler [46], World Wide Web Worm [37] and RBSE [16]. Googlebot [8] and Bingbot [9] are two of the most prominent examples of modern web crawlers. The on-going challenges in content crawling include the ever-increasing number of websites, increasing use of dynamic content, and the tension between crawling frequency for information freshness and demand on the Internet resources. Examples of studies that consider these problems include [6, 9, 12, 43]. Pandey and Olsten consider "user centric" web crawling in [44]. Their focus is on scheduling web crawls to specific pages in order to maintain the most up-to-date versions in search engine repositories. In [34] the authors performed large scale display ads data collection by developing a browser plug-in, their results showed that most of the ad categories are behaviorally targeted.

Several studies consider the problems associated with privacy and economics of online advertising. From an economics perspective, [23] studied companies advertising revenues as the function of user information, they showed that most of the advertising revenue was contributed by a small portion of users. [50] investigated the effectiveness of behavioral

[8] `http://support.google.com/webmasters/bin/answer.py-?hl=en&answer=182072`

targeting in commercial search engines. Several studies consider the problems associated with privacy and online advertising. Castelluccia *et al.* [11] demonstrated that one can reverse-engineer a user's profile by looking at targeted ads displayed to her and making inferences about the target interests revealed in the ads. Finally, the study by Guha *et al.* [25], which described challenges in measuring online advertising systems, informs our work.

3.6 Conclusions

In this chapter we describe a novel study of online display advertising *i.e.,* the Internet Adscape. The goals of our work are to develop a general understanding of the characteristics and dynamics of online ad markets. We perform the study on display ad as well as video ad markets. Our work begins by developing methods and tools for gathering display ads from a large number of web sites. Central to our ad-crawling method is the use of user profiles, which have an influence on the particular ads that are served. We collect and analyze ads shown to a variety of user profiles across the web and make a number of interesting observations. For instance, we find that while display ads are mostly targeted at the user profile, video ads are contextual. In the display advertising market we find that some profiles are targeted more than others. In the video ads market we find that the profile of the user gets changed significantly, even by video ads. These studies are the very first of their kind. They show that user experience of ads is dependent on many factors. We need to conduct more longitudinal analyses to enhance our understanding of ad delivery mechanisms and ad campaign dynamics on various markets. Finally, we need to drill down in greater detail on the mechanisms for ad targeting and build models that we can use to develop improved markets.

Chapter 4

Cardinal Auctions

User impressions present unique opportunities for advertisers to display their marketing messages. Every impression is characterized by an array of properties: the website on which it has occurred, what the user was doing before visiting that website, the user interests, etc. A single advertiser may differently value impressions that have occurred at different times but are identical otherwise. Auction is a standard way to sell unique goods such as user impressions.

4.1 Ad Auction Overview

Let there be a single user impression i for sale. There are n advertisers competing for it. Each advertiser j has a private valuation v_j, that is the value the advertiser receives if she wins the auction. Each advertiser j who is interested in impression i submits a bid b_j that indicates the maximum amount j is willing to pay in case of a win. As in any auction, two aspects are decided: (1) allocation — which ad will get shown; (2) pricing — how much the advertiser will pay.

Compared to traditional auctions, ad auctions are different in the following ways:

- auctioneer can charge advertisers for different *events*, e.g., impression, click, etc;
- ad auction can frequently be considered as a *repeated auction.* This is because a single publisher can have millions of user impressions daily.
- impressions are perishable, because if they are not sold at the moment of impression, they stay unsold.

While discussing advertisers, their interests are frequently presented through quasi-linear *utility function*:

$$u_j = \begin{cases} v_j - p_j & \text{advertiser } j \text{ wins the auction} \\ 0 & \text{otherwise} \end{cases} \tag{4.1}$$

Advertisers are strategic, and are interested in maximizing their utilities u's.

4.1.1 Single Item Auction

The literature presents us with multiple mechanisms to sell a single item.

First Price Auction. In the first price auction, the auctioneer has a single item to sell. There are n buyers (advertisers) competing for the item. Each participating buyer j submits her sealed bid b_j. Auctioneer receives all bids, and selects the maximum. The winner of the auction is the buyer with the highest bid $j^* = \arg\max_j b_j$ and she pays her bid b_j.

Buyers are strategic in this auction: if $b_j = v_j$, then $u_j = 0$. In order to maximize their utility, buyer strategy is not truthful — submit $b_j = v_j$.

Vickrey Auction. Bidders submit their sealed bids to auctioneer. [49] The highest bidder $j^* = \arg\max_j b_j$ wins and pays the second largest bid $p_j = max_{k,k \neq j} b_k$. The great advantage of this mechanism is that the advertiser's dominant strategy is to bid truthfully. The dominant strategy means that the bidder has a strategy that is at least as good as some other strategy. Dominant truthful strategy means that for the fixed bidder j, no matter what other bidders do, the utility of j is maximized when b_j is equal to v_j.

Equilibrium Concept

Assuming that there is no collusion between bidders, ad auction is a non-cooperative game. Each bidder's strategy is to maximize her utility. Formally, let b_j be a strategy of buyer a_j, and $u_j(b_j)$ be a utility she receives at a bid b_j. Then there is an equilibrium if the following statement holds for all advertisers:

$$\forall a_j \in A \; u_j(b_j^*) \geq u_j(b_j') \tag{4.2}$$

4.1.2 Cardinal Auctions

In cardinal auctions, there are n buyers competing for at most $m \leq n$ identical copies of an item in the auction. Each buyer wants to buy exactly one copy and has two *private* numbers v_i and k_i. The auctioneer has no prior information about the values of the buyers. The utility $u_i(v_i, k_i)$ that bidder i derives from the auction is

$$u_i(v_i, k_i) = \begin{cases} v_i x_i - p_i & \text{if number of copies sold is less than } k_i \\ -\infty & \text{otherwise} \end{cases}$$

where $x_i \in \{0, 1\}$ is an indicator variable that shows whether i was allotted a copy or not, and p_i is the price at which i obtains it.

Buyers express their preferences through *2 dimensional bid* (b_i, l_i) where b_i is the maximum amount buyer i is willing to pay if at most l_i copies are allocated. Note that b_i may differ from v_i, and l_i from k_i. Once the auctioneer gathers all the bids she has to decide on an optimal number of copies k^*, allocation of k^* copies according to function $a(.)$ and payments according to pricing function $p(.)$. In such mechanisms we will consider no bidder i will be a winner if $l_i < k^*$.

Motivating Scenarios. An important motivation arises in auctions for online ad markets. Ads shown on a single page compete for the user's attention, which depends on many factors such as the nature and format of the content of the page and those of the other ads on the page (including their placement, color, dynamic features, etc). To the first order approximation, the *number* of ads shown is an important feature. The number of ads on the page determined by the publisher of the web page. They choose the number of ads on a page based on a variety of techniques, such as machine learning or user studies. Advertisers cannot influence the number of ads shown on the page, hence bid based on the average of their values over the possible number of ads. Cardinal auctions are an alternative. They let advertisers explicitly specify how many other advertisers may appear with their ad on a given page.

Cardinal auctions are also suitable in a variety of other instances:

- Say we can produce a collector's item, such as a signed copy of an album or a book. The more exclusive the copy is, the more valuable it is to the possessor. How many copies shall we produce? While traditionally this is determined by estimating the demand function, one can imagine an auction-based method, where bidders can specify in some way the value of the item to them as a function of how many copies are made and sold.

- Consider a situation that arises in a data exchange such as BlueKai[1], where certain pertinent data about a user is sold for ad targeting. The data may be sold to any number of advertisers for targeting, but in some cases the more the information is shared, the less value it gives to the advertisers. Hence, when data is sold via auction, advertisers may wish to be able to influence how many others get access to the data.

Some auctions involve the sale of multidimensional resources (*e.g.,* number of items of type A and type B are sold). This more general setting leads to what we call *configuration auctions*. Formally, say there are d dimensional resources. In one example of a configuration auction, bidder i specifies two d dimensional vectors: vector $\mathbf{k_i}$ and $\mathbf{b_i}$, where $\mathbf{b_i}[j]$ is the maximum payment bidder i is willing to make for resource j, provided that not more than $\mathbf{k_i}[j]$ copies of resource j are sold in total, over all bidders. Such problems will arise in machine scheduling: some processes require more CPU cycles, while others need more memory, and each process may wish to have no more than a certain amount of memory allocated in total, since otherwise their performance will be affected. Another natural application arises in display advertising when the ads on a page may be of image, video or textual type, and bidders wish to have no more than a certain number of video ads on the page so that viewers are not distracted unreasonably. Cardinal auctions are a special case of configuration auctions, when $d = 1$. Of course, one can imagine bidding languages other than the example we have provided above with d-dimensional resources, so configuration auctions could be potentially even more general than cardinal auctions. We are not aware of any prior study of general d-dimensional configuration auctions, and as we show here, there are a number of interesting open issues, even with the study of cardinal auctions.

[1] `http://www.bluekai.com/`

4.2 Preliminaries

Allocation $\mathcal{A}$ is the set of k^* winners who obtain a copy. We consider set of *feasible* allocations $\mathcal{F}$: allocation $\mathcal{A}$ is *feasible* if $\{l_i \geq |\mathcal{A}|, k_i \geq |\mathcal{A}| : \forall i \in \mathcal{A}\}$. The total *efficiency* $\mathcal{E}_\mathcal{A}$ of allocation $\mathcal{A}$ is the sum of values of allotted bidders or $\sum_{i \in \mathcal{A}} v_i$.

There are two natural auctions to consider.

$\mathbf{VCG_{CA}}$. VCG_{CA} is a straightforward extension of the standard Vickrey-Clarke-Groves (VCG) auction [49, 13, 24]. VCG_{CA} is *truthful*, *i.e.,* bidders bid their true valuations. This is a well-known property of VCG mechanism. Thus we can assume that bidders submit their bids as (v_i, k_i). VCG_{CA} chooses the feasible allocation that maximizes the total sum of values: $\mathcal{E}^* = \max_{\mathcal{A} \in \mathcal{F}} \sum_{i \in \mathcal{A}} v_i$. Let $\mathcal{E}_{-i}$ be highest efficiency achievable without bidder i, then the price for bidder i is

$$p_i^{VCG} = \mathcal{E}_{-i} - \mathcal{E}^* + v_i \tag{4.3}$$

VCG_{CA}, as generally known, can have low revenue. Furthermore, with cardinality constraints its outcomes are not *envy-free*: a losing bidder would agree to purchase a copy for the price higher than what is being asked from winners. For more intuition, consider the following example:

Example 4.2.1. *There are 3 bidders with true valuations:* $A = (100, 1)$*,* $B = (90, 2)$ *and* $C = (80, 2)$*. For this setting* VCG_{CA} *will identify winning allocation with bidders* B *and* C *in it, since the total efficiency is* $90 + 80 = 170 > 100$*. It will charge* B *amount* $p_B = 100 - 170 + 90 = 20$ *and* C *amount* $p_C = 100 - 170 + 80 = 10$*. Thus total collected payment is 30. However, bidder A will envy this low payment of 30.* □

$\mathbf{MPP_{CA}}$. MPP_{CA} was introduced in [41] and is based on the *minimum pay property* for the outcome: auction requires every buyer to pay no more than what she would have bid, if she knew all other bids, to get the exact same assignment she got. To calculate prices, let winning allocation be $\mathcal{A}^*$ and let $\mathcal{A}_2$ denote the allocation that gives second highest sum of bids after $\mathcal{A}^*$. Let winning bids in $\mathcal{A}^*$ be sorted top-down in decreasing order of bids, $b_1 \geq b_2 \geq \ldots$. The ith winner pays price

$$p_i^{MPP} = \max\{\sum_{j \in \mathcal{A}_2} b_j - \sum_{j \in \mathcal{A}^*} b_j + b_i, b_{i+1}\}$$

The price consists of two components. The first term is the minimum amount i needs to bid to ensure that the allocation $\mathcal{A}^*$ is the winner, and the second term is the minimum bid to get above the $i+1$'st largest bid. The overall price is the maximum over both.

MPP auction is inspired by the *Generalized Second Price* (GSP) auction used by many popular search engines [14, 49, 5] to determine placement of advertisements on the page. In GSP there are n advertisers bidding for m advertisement slots. Each slot i has associated *click through rate* (CTR) with it, or probability of being clicked, denoted by $\alpha_i \in (0, 1)$. Slots are ordered in decreasing order of CTR's: $\alpha_i > \alpha_j$ for $i < j$. Advertiser i has private valuation v_i, which expresses the value of getting a click. To participate in the auction an advertiser submits bid b_i that indicates the maximum payment she is willing to make. The auctioneer receives all bids and assigns advertisers to slots in decreasing order of their bids. For convenience, we renumber advertisers in decreasing order of their bids, then advertiser i is assigned to slot i with CTR α_i. Payment of advertiser i is $p_i = \frac{\alpha_{i+1}}{\alpha_i} b_{i+1}$, and is charged only if the ad is clicked.

MPP_{CA} naturally generalizes GSP auction and in absence of cardinality constraints is the special case of GSP without click-through rates (when $\alpha_i = 1$ for all i).

Analysis of Auctions

Unlike VCG_{CA}, MPP_{CA} is not truthful. For example, a bidder can improve her utility by reporting $b_i \neq v_i$.

Example 4.2.2. *Consider 3 bidders with their true valuations:* $A = (100, 1)$, $B = (80, 2)$ *and* $C = (70, 2)$. *Auctioneer runs* MPP_{CA}. *If auctioneer receives truthful bids, then she will choose allocation of 2 bidders:* (B, C), *and charge them* $p_B = \max\{100 - 150 + 80, 70\} = 70$ *and* $p_C = \max\{100 - 150 + 70, 0\} = \max\{20, 0\} = 20$. *Utility of bidder* B *is* $u_B = 80 - 70 = 10$. *However, bidder* B *can improve it by lowering her bid to* $(40, 2)$. *Then, allocation is the same (bidders* B *and* C*), but payments are different: payment of*

B is 30, and payment of C is 60. Now, utility of B is $u'_B = 80 - 30 = 50 > 10$. *Hence, bidder B benefits from submitting untruthful bid* $b_B < v_B$. □

MPP_{CA} can have many outcomes, we consider only bid vectors that are in *Nash equilibrium*, that is, for every bidder $i \in \mathcal{A}$ the following inequalities hold:

$$v_i - p_{i,\mathcal{A}} \geq v_i - p_{j,\mathcal{A}'} \qquad \forall \mathcal{A}' \in \mathcal{F}, \mathcal{A} \neq \mathcal{A}'$$
$$v_i - p_{i,\mathcal{A}} \geq v_i - p_{j,\mathcal{A}} \qquad \forall j \neq i$$

where $p_{i,\mathcal{A}}$ denotes price for bidder i if allocation $\mathcal{A}$ is a winning allocation. There is a set of Nash equilibria efficiencies of $\Sigma_1, \Sigma_2, \ldots$. Define $\mathcal{E}_{min} = \min_i \Sigma_i$ and $\mathcal{E}_{max} = \max_i \Sigma_i$. Then, *price of anarchy* is defined as $PoA(.) = \frac{\mathcal{E}_{max}}{\mathcal{E}_{min}}$. VCG_{CA} is truthful, hence Price of Anarchy $PoA(VCG_{CA}) = 1$. Observe that maximum efficiency of MPP_{CA} $\mathcal{E}_{max}$ is bounded by highest efficiency $\mathcal{E}^*$ that is achieved by VCG_{CA}, or $\mathcal{E}_{max} \leq \mathcal{E}^*$. Thus, Price of Anarchy if MPP_{CA} can be bounded as follows:

$$PoA(MPP_{CA}) = \frac{\mathcal{E}_{max}}{\mathcal{E}_{min}} \leq \frac{\mathcal{E}^*}{\mathcal{E}_{min}}.$$

To analyze revenue we use VCG_{CA} as the benchmark for consistency, and compare it to the revenue collected by MPP_{CA}.

Bidders are strategic and their goal is to maximize their utility. Strategy s is *weakly dominant* if, regardless of what other bidders do, strategy s gets a player utility that is at least as high as utility obtained by playing any other strategy. Strategy s is *(strictly) dominant* if utility of playing strategy s is strictly larger than playing any other strategy, regardless of what other bidders do. We consider two types of bidders:

- *Conservative bidders* do not bid over their value, *i.e.,* $b_i \leq v_i$, hence they do not risk paying more than their true valuation and getting negative utility.

- *Rational bidders* can bid above their true valuation v_i in equilibria if and only if the payment p_i does not exceed v_i. Equilibria that contain such bids are fragile, because the bidder can get negative utility if some other bidder changes her bid. Such equilibria help us explore the properties of possible outcomes.

4.3 Efficiency and Revenue Analyses

Preliminary Observations

Our first observation is regarding truthfulness of the cardinal constraint.

Lemma 4.3.1. *In MPP_{CA}, bidders truthfully reveal their private k_i's, that is $l_i = k_i$ for all i in Nash equilibria of MPP_{CA} auction.*

Proof. Consider any bidder i who participates in winning allocation of size k^*. For contradiction let their revealed $l_i > k_i$. There are two possibilities: (1) $k^* \leq k_i$, then bidder i would have identical utility by reporting k_i instead of l_i; (2) $k^* > k_i$, bidder now has utility $-p_i$ which is worse than utility from reporting k_i truthfully. Hence, bidder does not have incentive to submit $l_i > k_i$.

Consider same bidder i, and say for contradiction that the bidder revealed $l_i < k_i$. Again, there are two cases: (1) $k^* \leq l_i$, then bidder i would have identical utility by reporting k_i instead of l_i; (2) with some positive probability $k_i \geq k^* > l_i$, bidder now has utility 0 which is worse than utility from reporting k_i truthfully. Hence, bidder does not have incentive to submit $l_i < k_i$. □

Our second observation concerns bidding behavior of *losing bidders*. There exist bids which are in equilibrium, but the efficiency is bounded away from the maximum achievable efficiency by an arbitrarily large factor. Consider the case of three bidders $(100, 1), (75, 2), (75, 2)$, who make the following bids respectively: $(100, 1), (1, 2), (1, 2)$. This set of bids forms a Nash equilibrium, since no bidder by herself has an incentive to change her bid. However, the efficiency of the resulting allocation is 100, compared to the optimum efficiency of 150. This is because losing bidders can arbitrarily shade their bids.

This is a common problem in presence of externalities. For example, [22] studies the advertising auction with two-dimensional bids for exclusive and nonexclusive display, and can have two types of outcomes: either a single ad is displayed exclusively, or multiple ads are simultaneously shown. Consider 3 bidders with valuations $(100, 0), (90, 80), (90, 80)$ respectively. Say they make the following respective bids:

$(100, 0), (1, 1), (1, 1)$. This set of bids forms a Nash equilibrium, and efficiency of allocation is 100, compared to the optimum efficiency of 160. Like in [22], we will henceforth assume that losing bidders bid their true valuation.

Efficiency

Theorem 4.3.1. *With conservative bidders, MPP_{CA}'s allocation has the same total value as VCG_{CA} in Nash equilibrium.*

Proof. Let $\sigma = \sigma_{VCG}$ be the set of winning bidders that maximizes efficiency and μ be the set of winning bidders under MPP_{CA} in Nash equilibria. Let $\{b_i | i \in \mu\}$ be a set of equilibrium bids under MPP_{CA}. Since MPP_{CA} chooses the set of bidders who maximizes total sum of bids, then it must be true that $\sum_{j \in \mu} b_j \geq \sum_{j \in \mathcal{A}} b_j$. Since σ is feasible, $\sum_{j \in \mu} b_j \geq \sum_{j \in \sigma} b_j$. Hence

$$
\begin{aligned}
& \sum_{i \in \mu \setminus \sigma} b_i + \sum_{i \in \mu \cap \sigma} b_i \geq \sum_{j \in \sigma \setminus \mu} b_j + \sum_{j \in \mu \cap \sigma} b_j \\
\implies & \sum_{i \in \mu \setminus \sigma} b_i \geq \sum_{j \in \sigma \setminus \mu} b_j \\
\implies & \sum_{i \in \mu \setminus \sigma} v_i \geq \sum_{j \in \sigma \setminus \mu} v_j \qquad (4.4) \\
\implies & \sum_{i \in \mu} v_i \geq \sum_{j \in \sigma} v_j
\end{aligned}
$$

In (4.4) we use the assumption on the right hand side that losers bid at least their true valuations and on the left hand side that bidders are conservative. Then the only possibility is that total value of μ equals that of σ. □

It follows that the PoA of MPP_{CA} is 1 for conservative bidders.

Theorem 4.3.2. *For rational bidders, PoA of MPP_{CA} is 2 and this is tight.*

Proof. Let μ and σ denote the set of bidders chosen by the allocation of MPP_{CA} and VCG_{CA} respectively. Let μ_2 denote the set of bidders who belong to the second best allocation of MPP_{CA}.

If $\sigma = \mu$, then the efficiency is 1 and we are done. Otherwise,

$$
\begin{aligned}
& & \mu = \sum_{i\in\mu} b_i > \sum_{j\in\sigma} b_j \\
\Longrightarrow & & \sum_{i\in\mu\backslash\sigma} b_i + \sum_{i\in\mu\cap\sigma} b_i > \sum_{j\in\sigma\backslash\mu} b_j + \sum_{i\in\mu\cap\sigma} b_i \\
\Longrightarrow & & \sum_{i\in\mu\backslash\sigma} b_i > \sum_{j\in\sigma\backslash\mu} b_j \\
\Longrightarrow & & \sum_{i\in\mu\backslash\sigma} b_i > \sum_{j\in\sigma\backslash\mu} v_j \qquad (4.5)
\end{aligned}
$$

where to get (4.5) we use assumption that losing bidders bid at least their value. Remainder of the proof deviates from the conservative bidder case as we cannot bound the left hand side like we did with conservative bidders.

Without loss of generality, assume that bidders in $\mu\backslash\sigma = \{b_1, b_2, \ldots, b_k\}$ are ordered in non-increasing order of bids, i.e., $b_1 \geq b_2 \geq \cdots \geq b_k$. To lowerbound the payment of the highest bidder, we start by working on one of the components of the pricing:

$$
\begin{aligned}
\sum_{i\in\mu_2} b_i - \sum_{i\in\mu} b_i + b_1 & \geq \sum_{i\in\sigma} b_i - \sum_{i\in\mu} b_i + b_1 \\
& \geq \sum_{i\in\sigma\backslash\mu} v_i - \sum_{2\leq i\leq k} b_i \qquad (4.6)
\end{aligned}
$$

where we get the first term of (4.6) from Eq. 4.5. Notice that if highest bidder i belongs to σ, then there is at least one bidder in μ who pays more then her value and gets negative utility. Hence, for allocation to be in Nash equilibria, the highest bidder i must belong to $\mu\backslash\sigma$, and we can exclude the highest bidder from the second term of (4.6). Hence,

$$
p(b_1) \geq \max\left\{ b_2, \sum_{i\in\sigma\backslash\mu} v_i - \sum_{2\leq i\leq k} b_i \right\}
$$

For other bidders, we bound MPP_{CA} payment by the $p(b_i) \geq b_{i+1}$. Using these, we get a lowerbound on the total revenue of MPP_{CA} as follows:

$$\begin{aligned}
\sum_{1\le i\le k} p(b_i) &= p(b_1) + \sum_{2\le i\le k} p(b_i) \\
&\ge \max\left\{b_2, \sum_{i\in\sigma\setminus\mu} v_i - \sum_{2\le i\le k} b_i\right\} + \sum_{2\le i\le k} b_{i+1} \\
&\ge \max\left\{b_2, \sum_{i\in\sigma\setminus\mu} v_i - \sum_{2\le i\le k} b_i + \sum_{3\le i\le k} b_i\right\} \\
&= \max\left\{b_2, \sum_{i\in\sigma\setminus\mu} v_i - b_2\right\} \ge \frac{1}{2}\sum_{i\in\sigma\setminus\mu} v_i \qquad (4.7)
\end{aligned}$$

Since, the bidders are rational their payments cannot exceed their valuations:

$$\sum_{i\in\mu\setminus\sigma} v_i \ge \sum_{i\in\mu\setminus\sigma} p(b_i) = \sum_{1\le i\le k} p(b_i) \qquad (4.8)$$

Combining (4.7) and (4.8) we get

$$\sum_{i\in\mu\setminus\sigma} v_i \ge \frac{1}{2}\sum_{i\in\sigma\setminus\mu} v_i. \qquad (4.9)$$

Then Price of Anarchy of MPP_{CA} is

$$\begin{aligned}
PoA(MPP_{CA}) &\le \frac{\mathcal{E}^*}{\mathcal{E}_{min}} = \frac{\sum_{i\in\sigma} v_i}{\sum_{i\in\mu} v_i} = \frac{\sum_{i\in\sigma} v_i}{\sum_{i\in\mu\setminus\sigma} v_i + \sum_{i\in\mu\cap\sigma} v_i} && (4.10)\\
&\le \frac{\sum_{i\in\sigma} v_i}{(1/2)\sum_{i\in\sigma\setminus\mu} v_i + \sum_{i\in\mu\cap\sigma} v_i} && (4.11)\\
&\le \frac{\sum_{i\in\sigma} v_i}{(1/2)\sum_{i\in\sigma\setminus\mu} v_i} && (4.12)\\
&\le 2.
\end{aligned}$$

We get (4.11) since for every i $v_i \ge 0$, and we get (4.12) since all $v_i \in \sigma\setminus\mu$ are also in σ.

Tightness. To see that the bound is tight consider three bidders with the following valuations: $A = (100, 1)$, $B = (50, 2)$, $C = (\epsilon, 2)$.

- One possible set of bids submitted can be as follows $b_A = (100, 1)$, $b_B = (100, 2)$, $b_C = (50, 2)$. The allocation with 2 bidders B and C yields more: $100 + 50 > 100$, hence MPP_{CA} will allocate bidders B and C, and will charge them 50 and 0 respectively. It is easy to check that none of the bidders can improve her utility acting on her own, hence the bids form Nash Equilibrium. Therefore $\mathcal{E}_{max} \ge 100$.

- Another possible set of bids submitted can be as follows: $b_A = (100, 1)$, $b_B = (50, 2)$, $b_C = (\epsilon, 2)$. In this situation, the single bid from A is higher than the sum of bids for two slot allocation: $100 > 50 + \epsilon$. Hence, MPP_{CA} will allocate single bidder A. Again, it is easy to verify that the bids form Nash Equilibrium. Thus $\mathcal{E}_{min} \leq 50 + \epsilon$.

Then $PoA(MPP_{CA}) = \frac{\mathcal{E}_{max}}{\mathcal{E}_{min}} \geq \frac{100}{50+\epsilon} \approx 2$. □

Revenue

Let $Rev(X)$ be revenue generated by mechanism X. We show two results.

Theorem 4.3.3. *With conservative bidders, $Rev(MPP_{CA}) \geq Rev(VCG_{CA})$.*

Proof. Let σ be the allocation with maximum total value (hence, the value attained by VCG_{CA}), μ be the allocation of MPP_{CA} in equilibrium, μ_2 be the set of bidders who participate in second best allocation of MPP_{CA} and σ_{-i} the set of bidders that gives the largest total value allocation when bidder i is not present.

Consider payments of each bidder $i \in \sigma \cap \mu$ under VCG_{CA} and MPP_{CA}:

$$p_i^{VCG} = \sum_{j \in \sigma_{-i}} v_j - \sum_{j \in \sigma} v_j + v_i$$

$$p_i^{MPP} = \max\{b_{i+1}, \sum_{j \in \mu_2} b_j - \sum_{j \in \mu} b_j + b_i\} \geq \sum_{j \in \mu_2} b_j - \sum_{j \in \mu} b_j + b_i$$

Since bidders are conservative and losers bid their values,

$$\begin{aligned}
p_i^{VCG} &= \sum_{j\in\sigma_{-i}} v_j - \sum_{j\in\sigma} v_j + v_i \\
&= \sum_{j\in\sigma_{-i}\setminus\mu} v_j + \sum_{j\in\sigma_{-i}\cap\mu} v_j - \sum_{j\in\sigma} v_j + v_i \\
&= \sum_{j\in\sigma_{-i}\setminus\mu} b_j + \sum_{j\in\sigma_{-i}\cap\mu} v_j - \sum_{j\in\sigma} v_j + v_i \\
&\leq \sum_{j\in\sigma_{-i}\setminus\mu} b_j + \sum_{j\in\sigma_{-i}\cap\mu} v_j - \sum_{j\in\mu} v_j + v_i \\
&= \sum_{j\in\sigma_{-i}\setminus\mu} b_j + \sum_{j\in\sigma_{-i}\cap\mu} v_j - \sum_{j\in\sigma_{-i}\cap\mu} v_j - \sum_{j\in\mu\setminus\sigma_{-i}} v_j + v_i \\
&= \sum_{j\in\sigma_{-i}\setminus\mu} b_j + \sum_{j\in\sigma_{-i}\cap\mu} b_j - \sum_{j\in\sigma_{-i}\cap\mu} b_j - \sum_{j\in\mu\setminus\sigma_{-i}} v_j + v_i \\
&\leq \sum_{j\in\sigma_{-i}\setminus\mu} b_j + \sum_{j\in\sigma_{-i}\cap\mu} b_j - \sum_{j\in\sigma_{-i}\cap\mu} b_j - \sum_{j\in\mu\setminus\sigma_{-i}} b_j + b_i \\
&\leq \sum_{j\in\mu_2} b_j - \sum_{j\in\mu} b_j + b_i = p_i^{MPP}
\end{aligned}$$

Now, consider payments of all such bidders $i \in \sigma\setminus\mu$ or $i \in \mu\setminus\sigma$. This is possible when there are 2 allocations of different size that have equally high efficiency. Let us denote them by $\mathcal{A}_\sigma$ and $\mathcal{A}_\mu$. If bidder i is present in only one of allocations, then her payment is $p_i^{MPP} = p_i^{VCG} = v_i$. Payment of VCG_{CA} follows from definition. Observe that bidder i submits a truthful bid in MPP_{CA}, because otherwise she would not be in the winning configuration. Now one can derive the payment from the definition. □

With rational bidders we show that revenue of MPP_{CA} can be as low as half of that of VCG_{CA}.

Example 4.3.1. *Consider 3 bidders with the following valuations* $A = (100+\epsilon, 1)$, $B = (50, 2)$ *and* $C = (50, 2)$. *Rational bidders can converge to bids* $A = (100, 1)$, $B = (100, 2)$ *and* $C = (50, 2)$ *respectively.* VCG_{CA} *gets truthful bids and chooses allocation consisting of bidder A, and her payment is 100, while* MPP_{CA} *chooses allocation with bidders (B and C), and prices them 50 and 0 respectively, achieving exactly half of revenue of* VCG_{CA}. □

4.4 Contrasts with Other Auctions

As mentioned earlier, without cardinality constraints MPP_{CA} becomes a GSP auction without click-through rates. It is known that in that case PoA of GSP is 1. To highlight our result further, we consider PoA of *prefix auctions* [4] and show that it is also 1.

The model is as follows. There are n ordered identical items to sell and m bidders, each bidder i has two private values v_i and k_i. Utility u_i of bidder i is $v_i - p_i$ if she obtains *any* of the first k_i copies, and it is $-p_i$ otherwise. Notice that now bidder i has positive utility, even if more than k_i copies are auctioned. In the auction, each bidder i submits a pair (b_i, l_i): b_i is the maximum they are willing to pay if they are allotted one of the first l_i copies.

We consider two different auctions: $pVCG$ and $pGSP$:

pVCG. $pVCG$ is the extension of VCG and is truthful. In the auction bidders submit their true valuations (v_i, k_i) to the auctioneer. Upon receival of bids the auctioneer creates a feasible allocation that maximizes total efficiency and calculates payments using Eq. 4.3.

pGSP. $pGSP$ is an iterative second price (SP) auction from the first copy of the item to the last, where for each item we run a SP auction among bidders who are not yet assigned a copy, but who still have nonnegative utility from obtaining one.

Notice that a bidder cannot benefit by submitting bid $b_i > v_i$, hence it is a weakly dominant strategy for bidders to bid *conservatively.* Similarly to MPP_{CA} it is a dominant strategy for the bidder to report her preference k_i truthfully. The argument is identical to Lemma 4.3.1.

Unlike second price auction, $pGSP$ is not truthful. Thus, similarly to MPP_{CA}, we analyze bid vectors that are in Nash equilibria. For $pGSP$, Nash equilibrium is defined as follows. For each i,

$$v_i - p_i \geq v_i - p_j \ \forall j \neq i \text{ and } j \leq \min\{k, k_i\}$$

Here we give PoA results for prefix auctions without click-through rates.

Theorem 4.4.1. *PoA of $pGSP$ is* 1.

Proof. Let σ be a set of bidders allocated by $pVCG$ and δ be a set of bidders allocated by $pGSP$ in Nash equilibria.

Consider total valuations of bidders that belong to only one of two allocations:

$$\sum_{i \in \sigma \backslash \delta} v_i \quad \geq \quad \sum_{j \in \delta \backslash \sigma} v_j$$

It is possible only if, among bidders from $\sigma\backslash\delta$ and $\delta\backslash\sigma$, the bidder with maximum valuation is in σ, or

$$\exists i \in \sigma\backslash\delta \; \forall j \in \delta\backslash\sigma \text{ s.t. } v_i > v_j \tag{4.13}$$

If it is not the case, then the maximum must be in $\delta\backslash\sigma$: $\exists j \in \delta\backslash\sigma \; \forall i \in \sigma\backslash\delta$ s.t. $v_j > v_i$. However, this is possible only if bidder v_j cannot replace any of bidders $i \in \sigma\backslash\delta$, otherwise v_j would improve $\mathcal{E}^*$. This, in turn, is possible if and only if $|\{i|i \in \sigma\backslash\delta\}| = 0$ that would imply that $|\sigma| < |\delta|$. However, it leads to a contradiction, as efficiency of $\mathcal{E}^*$ could be improved by adding v_j to it.

If (4.13) is true, then there must be a losing bidder l who can raise her bid, enter the allocation and as the result improve her utility. That creates a contradiction. Thus, total values of σ and δ are identical.

□

It is believed that one of the reasons to use MPP auctions is to improve revenue, *e.g.,* revenue of GSP without click-through rates is always at least as much as that of VCG. This is also true for modification presented in [22]. Likewise, for prefix auctions, this continues to hold.

Theorem 4.4.2. *In equilibrium, $Rev(pVCG) \leq Rev(pGSP)$.*

Proof. Let $\mathcal{A}$ be the allocation of $pGSP$ (or $pVCG$). Consider payment of bidder $i \in \mathcal{A}$. Let l be the bidder who enters allocation $\mathcal{A}$ if i leaves it. If no such bidder exists, let l be a bidder with valuation $v_l = 0$ and $k_l = n$. Then payment of bidder i in $pVCG$ is $p_i^{pVCG} = \mathcal{E}_{-i} + \mathcal{E} + v_i = v_l$ and payment of bidder i in $pGSP$ allocation $p_i^{pGSP} = \max\{b_{i+1}, b_l\}$.

Payment is minimized when $p_i^{pGSP} = b_l$. Also, $b_l = v_l$, since bidders are conservative, and l is losing bidder. Hence, $Rev(pVCG) \leq Rev(pGSP)$.

$\square$

In contrast to $pVCG$ and VCG, which have lower revenue than the corresponding versions of MPP, for cardinal auctions we have shown that in some cases VCG_{CA} may have more revenue than MPP_{CA}.

4.5 Concluding Remarks and Future Directions

We consider the problem of selling identical copies of an item via an auction in which the number of copies sold is unknown *a priori*, and valuation of a bidder depends on the total number of winners. This scenario is motivated by the number of ads on a page or number of parties that get access to certain information. While there are many ways to solve this problem, we consider *cardinal auctions*, in which the bidding language lets buyers explicitly bid on the maximum number of winners allowed. Our work analyzes cardinal auctions of MPP_{CA} and VCG_{CA} for revenue and efficiency tradeoffs in equilibria, and shows that they are quite different from the case without the cardinal externality. We find that MPP_{CA}, which is inspired by the widely used Generalized Second Price (GSP) auction, has surprising properties. In the case of rational bidders, efficiency of MPP_{CA} is half of that of VCG_{CA}. At the same time, in the worst case MPP_{CA} can collect only half of revenue of VCG_{CA}.

There are many open directions to pursue. For example, in display ads, slots may differ in terms of their location, dimensions and click-through rates. We need to extend the study of cardinal auctions to auctions for configurations of display ads with varying quality scores, or with varying click through models.

Externality is a richer phenomenon than we have studied here. For instance, the value for a bidder might depend not only on the number of other possessors, but also on their identity, quality, *etc.* Further, one can consider bidding languages which go beyond the step function we have adopted here, for example, by letting bidders specify their value for each potential number of winners. Studying such notions of externalities and bidding languages is an active area in economics, and problems are still open.

From a technical point of view, we would like to extend our analysis to Bayesian case and study dynamics of cardinal auctions.

Chapter 5

Online Ad Allocation with Secondary Metrics

Online matching and ad allocation is a topic that has recently received considerable attention, due to its importance in optimization of online ad markets, and because this is an interesting research field in itself. The matching was introduced in [29]. In a classical problem, there is a bipartite graph $G(\mathcal{J}, \mathcal{I}, E)$. Vertices $\mathcal{J}$ are known in advance, and vertices $\mathcal{I}$ arrive online, one after another. Upon arrival of vertex $i \in \mathcal{I}$ its adjacent nodes in $\mathcal{J}$ are revealed, and it has to be matched to one of them. Each match is irrevocable. The goal is to maximize the total number of matches.

In the language of online ad markets this model can be said as follows. There is a bipartite graph $G(\mathcal{J}, \mathcal{I}, E)$, in which $j \in \mathcal{J}$ represent ads; j's are known. User impressions $i \in \mathcal{I}$ arrive online. Upon arrival, a set of eligible ads (adjacent nodes in $\mathcal{J}$ for i) is revealed. The impression has to be matched to one of the eligible ads. The match is irrevocable, that is once the match is made and the ad is shown to the user it cannot be undone. The goal is to maximize the total number of ads matched to impressions.

Inspired by online ad allocation, many models and results have been developed for online matching problems. There are number of generalizations, most common involve some of the following:

- *weights* on edges, meaning edges can vary in their quality;
- offline nodes may have *bids*, the bid is the maximum amount the ad is agreeing to pay for the impression;
- offline nodes may have *capacity*, the capacity denotes the maximum number of times the ad can be matched;

- offline nodes may have *budget*, the budget is the maximum total payment the ad is willing to make.

Most of the previous work deals with optimization of a single objective. It is frequently either cardinality of the matching (*i.e.,* number of matches) or its weight (*i.e.,* sum of weights of edges matched). Recently, there has been new work done on online ad allocation problems with multiple contradicting objectives [3, 31]. The models consider formulations in which objectives of the allocating platform and advertisers are not aligned. Further, the advertiser's goal in the market is presented to be one dimensional. For instance, in display ad market advertisers are interested in maximization of a number of impressions, and in search ad market advertisers are interested in maximization of a number of clicks. However, in practice, advertisers are interested in some combination of impressions, clicks and conversions. In this chapter we study a formulation with two objectives, one of which is *targeted.* We are using in-app advertising market as an illustrative example.

In particular, we consider a setting in which a set of fixed nodes (ads) in addition to bids b_j and budgets B_j have cost-per-acquisition (CPA) targets w_j's. Upon arrival of an online item i a set of eligible fixed neighbors (ads) for the item is revealed together with pairwise weights CTR_{ij} and CVR_{ij} for eligible neighbors j's. The problem is to assign each item i to an eligible neighbor online, while respecting the budgets and CPA targets. The goal is to maximize the total weight of the matching based on CTR weights while meeting CPA targets. This is an optimization problem, in which one metric (CTR based weight of the matching) is optimized and the other is targeted (empirical CPAs cannot exceed CPA targets). One could think, that this problem may be written as a two parameter optimization problem, where the first objective is to maximize the total CTR-based weight of the matching and the second objective is to minimize average cost per acquisition. However, the solution for this bicriteria problem is not necessarily feasible for the initial problem. In particular, a CPA target w_j specified by the advertiser may be exceeded. It could be the case, that deviation from the targeted metric can be incorporated into the objective function as a *penalty.* Here the challenge is to find a suitable penalty function. In Section 5.6 we discuss aspects that have to be captured by the penalty function. Our approach is to reduce the problem to one dimension optimization by converting the second criteria into terms of the CTR weight of the matching.

5.1 Mobile App Install Business

In this chapter we use the mobile app install business as an example of the market where the phenomenon of the secondary targets presents itself. It is the market of application ads displayed in mobile applications. Ads that promote mobile applications are frequently shown in ad slots of other mobile applications, because that is the only way to know that the user who sees the ad has a qualifying mobile device. This is a rising business, as user time is increasingly spent on mobile devices. As mobile devices become an everyday norm, more and more apps appear. We increasingly use apps for shopping, reading, listening to music, communicating with friends, managing our documents and lives, logging our workouts, tracking caloric intake, *etc.* There is an app for everything. The number of unique users of apps has eclipsed the television audience by size and time spent[1]. According to Flurry Analytics, in 2014 overall app usage grew by 76%[2]. App developers increasingly turn to advertising to promote discovery, popularize and distinguish their apps in overwhelmingly large stores of millions of apps.

On the high level, the structure of the market is as follows. There are advertisers. Advertisers create ad campaigns $j \in \mathcal{J}$. The conversion event of an application ad is an installation of the application or *"install"* and the goal of such an ad is acquisition of a new user. Most attributes of ad campaigns are standard: ad j has a daily budget B_j, a bid b_j, targeting criteria, frequency caps, creative, *etc.* In addition, an advertiser has a clear understanding of the maximum amount she wants to pay per conversion, on average. We refer to this amount as *CPA target* (cost per acquisition target) and denote it by w_j.

There are publishers. As we have mentioned publisher are mobile applications.

---

[1] `http://marketingland.com/report-time-on-mobile-devices-surpassed-tv-108366`

[2] `http://flurrymobile.tumblr.com/post/115194992530/shopping-productivity-and-messaging-give-mobile`

Publishers are calling the market to get ads for impressions. Their goal is to maximize their revenue.

Advertisers are interested in installs, however they are rarely paying *per conversion.* It is risky for the serving platform to operate on a conversion level, *i.e.,* charge advertisers per conversion. As in any other online ad market, conversions are rare events and hence are very hard to predict. It is also risky to pay publishers on *per impression* basis because then publishers may send low quality impressions to the market. Therefore, platforms often choose to operate on *pay-per-click* or *pay-per-completed view* models. Clicks and completed views happen more frequently, hence are easier to predict, compared to conversions. But in order for markets to be successful long term, markets need a better performing conversion optimization to meet the price per conversion app advertisers have in their minds. Notice, that b_j and w_j can be two independent numbers. In addition w_j are not necessarily realistic or achievable on the market.

5.2 Allocation with Secondary Metrics

Application ads serving systems assign ads to apps, on behalf of publisher applications respecting targeting criteria and delivery goals of advertisers (budgets B_j's and CPA targets w_j's). Publishers want to optimize overall quality, measured by clicks. Therefore a desirable property of an ad-serving system is to maximize this quality while satisfying requirements of ad campaigns. However, the allocation optimizing quality may not be satisfactory for a given set of w_j's. This motivates our model of the ad problem as maximizing weight (or revenue) and meeting CPA targets w_j's.

5.2.1 Modeling the Problem

More specifically, we are studying the following online allocation problem: let $\mathcal{J}$ be the set of advertisers, and $\mathcal{I}$ be the set of impressions. Each advertiser $j \in \mathcal{J}$ comes with a bid b_j— the amount j is agreeing to pay per click on an impression, and daily budget B_j— the maximum amount of payments j is willing to pay in one day. The platform can choose different ads for each impression $i \in \mathcal{I}$. The eligibility of ads to impressions is represented by a bipartite graph $G(\mathcal{I}, \mathcal{J}, E)$, where $(i, j) \in E$ if and only if ad j is eligible to be shown to impression i. We denote by $\mathcal{N}(\cdot)$ the neighborhood of a node in G. Note that G is undirected and $i \in \mathcal{N}(j) \iff j \in \mathcal{N}(i)$. Each pair $(i, j) \in E$ is associated with the following probabilities, that may not be a priori known, but, for instance, can be estimated:

- CTR_{ij}– the probability of a click on ad j shown at impression i;
- CVR_{ij}– the probability that, given a click, the user converts (*e.g.,* installs the app).

When impression i arrives, it has to be matched to an ad $j \in \mathcal{N}(i)$, or left unmatched. The match, once made, is irrevocable. The ad is charged $p_{ij} \leq b_j$ in the event of the click.

Definition 5.2.1. ***Feasible Allocation*** *Let w_j denote the target CPA of ad j and $click_{ij}$ be a binary variable that indicates whether click on ad j shown on impression i occurred (=1) or not (=0). Let $\omega_j = \frac{\sum_{i \in \mathcal{N}(j)} click_{ij} \times p_{ij}}{\#conversions}$ be the empirical average cost of acquisition observed by the ad j. Notice that values of ω_j's are observed values and can be interpreted as follows. Let the platform run for let's say a day, and at the end of that period for each ad j the following inequality must hold:*

$$\omega_j \leq w_j, \quad \forall j \in \mathcal{J}. \tag{5.1}$$

We refer to allocations that satisfy Eq. 5.1 as feasible.

5.3 The Global Optimum. The Platform Perspective.

Assuming complete knowledge of the data, a theoretical benchmark for any budget allocation algorithm can be obtained via integer programming [2]. For a given linear objective we can attempt to discover the best allocation policy using integer of linear programming from a perspective of the ad serving platform. There are several linear programming formulations. Here we discuss two. The objective function of the first formulation is to optimize the total number of expected clicks. The requirement for

CPA targets is written as the constraint in the problem. The second formulation has a distinct objective function that aims at optimization of a linear combination of expected clicks and expected conversions.

Maximizing Total Number of Clicks. We can discover the best allocation policy that maximizes total clicks as the solution of the following linear program:

$$\begin{aligned}
\text{maximize} \quad & \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{N}(i)} x_{ij} CTR_{ij} & \\
\text{subject to} \quad & \sum_{i \in \mathcal{N}(j)} x_{ij} CTR_{ij} b_j \leq \sum_{i \in \mathcal{N}(j)} x_{ij} CTR_{ij} CVR_{ij} w_j & \forall j \in \mathcal{J} \\
& \sum_{i \in \mathcal{N}(j)} x_{ij} CTR_{ij} b_j \leq B_j & \forall j \in \mathcal{J} \\
& \sum_{j \in \mathcal{N}(i)} x_{ij} \leq 1 & \forall i \in \mathcal{I} \\
& x_{ij} = \{0, 1\} & \forall (i, j) \in E
\end{aligned} \tag{5.2}$$

In this linear programming the goal is to maximize number of expected clicks, such that the expected spend is below advertisers' budgets, and expected CPA is below targets.

Maximizing Total Clicks and Minimizing Empirical CPA's. Eq. 5.2 works under the assumption that expected CPA cannot exceed the target. However, in reality the line between acceptable CPA and unacceptable is blurred. For instance, there can exist a solution that (in expectation) delivers a large number of conversions and exceeds the target CPA w_j by one cent. Clearly, there are two competing objectives: maximization of total number of clicks and minimization of average cost of conversion. Below is one possible formulation of integer programming with an objective function that maximizes the sum of number of expected clicks and conversions per dollar spend.

However, observe that value of w_j's can vary from one ad campaign to another.

$$
\begin{aligned}
\text{maximize} \quad & \alpha(\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{N}(i)} x_{ij} CTR_{ij}) + \beta(\sum_{j \in \mathcal{J}} \frac{\sum_{i \in \mathcal{N}(j)} x_{ij} CTR_{ij} CVR_{ij}}{\sum_{i \in \mathcal{N}(j)} x_{ij} CTR_{ij} CPC_{ij}}) \\
\text{subject to} \quad & \sum_{i \in \mathcal{N}(j)} x_{ij} CTR_{ij} CPC_{ij} \leq \sum_{i \in \mathcal{N}(j)} x_{ij} CTR_{ij} CVR_{ij} w_j && \forall j \in \mathcal{J} \\
& \sum_{i \in \mathcal{N}(j)} x_{ij} CTR_{ij} CPC_{ij} \leq B_j && \forall j \in \mathcal{J} \\
& \sum_{j \in \mathcal{N}(i)} x_{ij} \leq 1 && \forall i \in \mathcal{I} \\
& x_{ij} = \{0, 1\} && \forall (i, j) \in E
\end{aligned}
\tag{5.3}
$$

This is a bicriteria optimization problem. Its two objective functions are not necessarily aligned. For instance, the solution that maximizes the expected number of clicks is distinct from the one that maximizes the cost per conversion. One solution would be to search for an allocation with the optimal click-conversion trade-off, or Pareto optimal [45]. In Related Work, Section 5.5 we discuss [3, 31] and their applicability to allocation optimization with a secondary targets problem.

Solving the integer programming would find the optimal allocation from a perspective of the platform. However, a number of assumptions need to be made. First, the formulation relies on the fact that the empirical CPA will exactly match the expected. That is hardly the case, with empirical conversion rates of 0.007, given a completed view. Second, we also need to assume that predictions CTR_{ij} and CVR_{ij} are correct, and that a set of impressions is constant day after day (*i.e.,* impressions are sampled from a fixed distribution that does not change over time). Finally, integer programming will find an optimal, but not necessarily fair allocation. For instance, the solution will allocate set of impressions to ad j, because such allocation is, in general, better for the platform, however, given an opportunity, ad j would choose a different set of impressions for itself.

One approach to ad allocation is repeated auctioning with some budget admission control. The auction happens in following stages:

1. Impression i arrives to the platform;

2. The platform retrieves set of eligible ads $\mathcal{N}(i)$ using information of the user, the publisher, etc.;

3. The platform ranks eligible ads using some *ranking function*;

4. The ad j^* that gets the highest rank wins the auction and is allocated for the impression i.

Further, we consider approaches that affect a repeated auction used for ad selection. One can affect the outcome of the auction in two principal ways:

- through modification of ranking function of the auction.

- through altering the set of ads that participate in the allocation.

Of cause one can use a combination of both approaches.

5.3.1 Ranking Based Algorithms

The selection of an ad for the impression at the serving platform is performed in real-time. When a request for an ad arrives, the set of eligible ads is generated. This set takes into account properties of impression, user and ads available for serving. Then ads are ranked using some *ranking function*. The ad that is ranked first is chosen and shown to the user. This is a local greedy approach. For instance, in search ads eligible ads are ordered in decreasing order of their expected revenue. For a given (impression, ad) pair the score is calculated as $s_{ij} = CTR_{ij}b_j$. Clearly, by varying the coring function, one can steer the properties of the allocation in different directions. For instance, instead of revenue, one could choose to optimize greedily for conversions and ranking by $s_{ij} = CVR_{ij}$. Below we introduce two ranking scores that are inspired by formulations of linear programmings in Eq. 5.2 and Eq. 5.3.

Linear Combination of Revenue and Conversions

Consider Eq. 5.3. The objective function has two clear parts: one is expected number of clicks, and the second is expected price per conversion. In other words, for each advertiser we want to increase her spend (maximize $CTR_{ij}b_j$) as well as deliver a larger number of conversions (maximize $\sum_i x_{ij}CTR_{ij}CVR_{ij}$). To direct the allocation in that process we combine these terms into a single scoring function. Notice that the former is in dollars and the latter is a probability. We build a scoring function that is a linear combination of two normalized values:

$$s_j^{lin} = \alpha \times norm(CTR_{ij}b_j) + \beta \times norm(CVR_{ij}) \tag{5.4}$$

where $norm(CTR_{ij}b_j) = \frac{CTR_{ij}b_j}{\max_{j\in\mathcal{N}(i)} CTR_{ij}b_j}$ and $norm(CVR_{ij}) = \frac{CVR_{ij}}{\max_{j\in\mathcal{N}(i)} CVR_{ij}}$.

Credit CPA

Consider Eq. 5.2. Here we want the advertiser to participate in impressions s.t., at the end of the day the CPA targets are greater than or equal to observed values. In other words, as impressions i arrive online, the platform needs to

- optimize its revenue, as well as
- aim to meet w_j's of ads at the end of the day.

Consider allocation of impression i to ad j. Two things happen in expectation: 1. ad network gets α_{ij} of revenue, and 2. advertiser gets CVR_{ij} of conversions. Given w_j, the advertiser's expected number of conversions is $\frac{\alpha_{ij}}{w_j}$. That creates a deficit of $\frac{\alpha_{ij}}{w_j} - CVR_{ij}$. The current expected price for it is $\alpha_{ij} \times CVR_{ij}(\frac{\alpha_{ij}}{w_j - CVR_{ij}})$. Hence, resulting expected revenue is

$$rev_j = \alpha_{ij} - \alpha_{ij} \times CVR_{ij}(\frac{\alpha_{ij}}{w_j} - CVR_{ij}) \tag{5.5}$$

$$= \alpha_{ij}(2 - \frac{\alpha_{ij}}{CVR_{ij}w_j}) \tag{5.6}$$

where $CVR_{ij}w_j$ is cost of CVR_{ij} of conversions.

We use rev_j's to rank ads and allocate the ad that maximizes the score $s_j^{credit} = rev_j$. If the quality of arriving impressions is uniform or comparable, and all the predictions are correct, then in expectation ads j's will achieve their corresponding w_j's.

5.3.2 Throttling Based Algorithm

Optimal for a Single Advertiser. Consider the problem from a single advertiser j's perspective. Then our goal is not only to optimize the revenue of the platform, but also meet secondary target w_j. It can be the case, that advertiser j participates in all impressions for which j is eligible and doesn't meet her target w_j. Then one solution to improve j experience is let j to participate in some subset of impressions and *throttle* j from others.

More formally, the CPA target bidding problem is as follows. Impressions $i \in \mathcal{I}$ arrive online, and for each arriving impression, we need to decide whether we bid b_j on behalf of ad j or not. For the impression i and ad j pair we have a probability CTR_{ij} of getting a click, and its cost CPC_{ij}. It also has some probability CVR_{ij} of conversion, conditioned on the click. The problem is to spend budget B_j and maximize the total expected number of conversions:

$$\begin{aligned} \text{maximize} \quad & \sum_{ij} x_{ij} CTR_{ij} CVR_{ij} & \\ \text{subject to} \quad & \sum_{i} x_{ij} CTR_{ij} CPC_{ij} < B_j & \\ & \sum_{i} x_{ij} \leq 1 & \forall i \\ & x_{ij} \in \{0,1\} & \forall i \end{aligned} \tag{5.7}$$

This is the knapsack problem [36], where participation in impression i has weight $CTR_{ij}CPC_{ij}$ and profit $CTR_{ij}CVR_{ij}$. Then, to solve Eq. 5.7, we pick impression i in decreasing order of $\frac{CVR_{ij}}{CPC_{ij}}$ until B_j is spent.

Further, to meet CPA target w_j of ad j, we have to pick a threshold τ_j, for instance, based on historic data, and bid on all auctions where $\frac{CVR_{ij}}{CPC_{ij}} \geq \tau_j$. For more details see Algorithm 2.

Optimal For Multiple Advertisers. Now let us consider the case for N advertisers. Assuming that all advertisers are playing the strategy described above, intuitively, it is possible that probabilities for clicks and conversions are correlated across different ads for a single impression. For instance, if user is in the market for a new game application, the user can click on a number of ads of gaming applications. Here the event of

Algorithm 2 Setting τ_j for j

```
1: procedure SetThreshold(I, w_j, B_j)
2:     Rank impressions i ∈ I in decreasing order of CVR_ij / CPC_ij
3:     Pick the top impressions I_j in I according to the ranking such that
4:         the budget runs out: Σ_i x_ij CTR_ij CPC_ij < B_j
5:       and
6:         the CPA target is satisfied: (Σ_i x_ij CTR_ij CPC_ij) / (Σ_i x_ij CTR_ij CVR_ij) ≤ w_j
7:     Find τ_j
8: end procedure
```

the click is conditioned on the fact that the user is searching for the app. Then it can be the case that independently computed thresholds τ_j's are preventing advertisers from spending their budgets because of the competition. Clearly, τ_j's have to be adjusted to take into account the competition for impressions. Consider Algorithm 3.

Algorithm 3 Algorithm Choose Thresholds for All Advertisers

```
1: for j ∈ J do
2:     Find τ_j, given I, J
3: end for
4: If τ_j's did not converge, go to Step 2.
```

In practice, this approach can be solved in an ad hoc manner. One can simply guess and manage τ_j's from one day to another. This can be achieved using control loop feedback mechanism [47]. However, this approach will create a higher competition for impressions of higher value, and lower to no competition for lower value impressions. As a result the price-per-conversion can go up (due to the higher competition), and the number of impressions served will go down, leaving some impressions unfilled. This can result in lower revenue for both publishers and the platform.

5.3.3 Experimental Results

We report experiments we have performed live on on Flurry traffic. We start with a description of live experiments, and follow with an analysis performed on data.

Live Experiments

We conducted a line of experiments on live traffic of Flurry Ad Network in July 2014. At the time, Flurry was an independent company running multiple lines of business. One of Flurry's functions was running an application ads serving system. It was a marketplace in which publishers were app developers who monetized their apps and users through advertising. In particular, these were applications that were showing video ads. The demand side was formed by advertisers promoting their apps through video ads.

The market was set up as follows. Advertisers created ad campaigns j, each had bid b_j— the maximum amount the advertiser was willing to pay for *completed view* of the ad, daily budget B_j— the maximum total amount of money the advertiser was willing to pay on a given day, and CPA target w_j— the maximum average payment per conversion (installation of the app in this particular case) the advertiser was aiming at. Each ad also had targeting criteria describing the segment of the market of interest, frequency caps, and other criteria. The advertiser's goal in the market is best described as "spend the daily budget while still meeting CPA target." Publishers were accessing the Flurry market upon impression arrival, requesting ads. Publishers were paid for each completed view. Flurry was paid a fixed percent of the payment per completed view.

Data and Setting

By default, each impression was served in the following steps: (1) Flurry ad server receives ad request with accompanying request attributes (*e.g.,* user gender, age, location, publisher app, etc); (2) Flurry ad server retrieves the set of eligible ads; (3) Ads are ranked in decreasing order of expected revenue; (4) The highest ranked ad is returned to the publisher, and the advertiser account is charged b_j in each case of a completed view.

When running experiments, we have created a number of treatments. Each treatment T_k was implementing a distinct algorithm. The treatment was appointed to serve a random fraction of impressions r_k. The default ad serving process was executed with probability $d = 1 - \sum_k r_k$. Upon arrival of an impression we tossed a coin to choose the algorithm to use for the serving. Altogether, it worked as shown in Algorithm 4.

Algorithm 4 Running Experiments on Live Traffic.

procedure ServeImpression($i, \{< T_k, r_k >\}$)
  $d = 1 - \sum_k r_k$ - default serving of i
  Sample from multinomial $(\{r_k\}, d)$ to determine the algorithm to use for the i
  Serve i using the chosen algorithm.
end procedure

To track performance of our experiments we had access to the following aggregates that were computed daily for each treatment as the default setup: (1.) number of impressions per (ad space, ad) pair; (2.) number of completed views per (ad space, ad) pair; (3.) number of conversions per (ad space, ad) pair. Notice that conversion information was not perfect, and came in with the delay. We registered the conversion event in one of two ways: (a.) The advertiser would inform us of the number of conversions on the given day. This information usually came from third parties that were tracking performance of the advertiser's ads. (b.) If an advertiser's ad was instrumented with Flurry analytics code we would learn about the conversion event after the user opened the app. In other words, the conversion was an event of the user opening the app for the first time, within 3 days of seeing the ad.

Algorithms Tested

In previous sections of the chapter we have discussed details of the problem as well as potential approaches to the solution. In this section we implement and test on live traffic as discussed in Section 5.3.3, in particular we test approaches which are based on knapsack and two formulations of the ranking function. We start with a description of the default ranking function used to order ads before selection of the winning ad. Then we describe novel approaches to ranking. In order to make revenue across multiple treatments comparable we charged winning ad its bid, *i.e.,* we were running a first price auction.

a. Default Ranking: Expected revenue. Allocating the impression i to the ad that maximizes expected revenue is standard in the industry. If estimates CTR_{ij}'s for

$j \in \mathcal{N}(i)$ are good, then this allocation rule is to maximize revenue. Ads $j \in \mathcal{N}(i)$ are sorted in decreasing order of

$$s_j^{ecpm} = CTR_{ij} b_j.$$

Impression is allocated to the ad with the largest score.

b. Linear Scoring Function. In this treatment we scored eligible ads using the scoring function described in Eq. 5.4. α and β were set to 1 in initial experiments.

c. Credit CPA Scoring Function. We use Eq. 5.6 ranking function to rank eligible ads and find the winner.

d. Linear Scoring Function with Daily Updated Per Ad Thresholds. As discussed in Section 5.3.2, one way to satisfy constraint that makes sure that advertisers meet their w_j's is to allow j to participate in allocation of impressions of necessary quality per cost. Notice that, in the case of Flurry, which was charging advertisers their bids, the algorithm simplifies even further. For details of the algorithm for a fixed ad campaign j see Algorithm 5.

Algorithm 5 Setting τ_j for j

```
procedure ChooseThresholdForAd(N(j), w_j, B_j)
    Rank all impressions i ∈ N(j) in decreasing order of cost-per-conversion CVR_ij
    Take first k impressions (N(j)^k) such that:
        the budget runs out: Σ_{i∈N(j)^k} x_ij CTR_ij b_j < B_j
      and
        the CPA target is satisfied: Σ_{N(j)^k} CTR_ij b_j ≤ w_j × Σ_{N(j)^k} CTR_ij CVR_ij
    Set τ_j to minimum value of CVR_ij of ad in N(j)^k
end procedure
```

There is a straightforward way to transform this to an online method:

1. (Initialization:) we establish thresholds τ_j's, s.t., ads only participate in impressions with $CVR_{ij} \le \tau_j$. The initial values of τ_j's can be calculated as follows:

$$\tau_j = \frac{b_j}{w_j} \tag{5.8}$$

2. The initialized value may not be optimal, because: (a) if several advertisers are executing the packing strategy, they can be packing the same set of impressions; (b) predictions are frequently imperfect; (c) if j participated only in allocation

of impressions with $CVR_{ij} \geq \tau_j$, then it's empirical CPA $\omega_j = \frac{\#clicks \times b_j}{\#conversions}$ will be lower then w_j and due to the fact that j doesn't win all of the impressions its spending and number of impressions will go down. (d) the traffic differs day to day. That will introduce inefficiencies. Hence, we want to perform periodic updates of τ_j's. The simplest update is additive δ: for each τ_j we (a) raise it by δ if a_j exceeds its w_j, and (b) lower otherwise.

The application of scoring function and pricing is as follows. When impression i comes in, we compute the score according to $s_j^{(.)}$ for all eligible ads $j \in \mathcal{N}(i)$. The impression is allocated to the ad j that maximizes the score. If scoring function uses thresholds, then each $j \in \mathcal{N}(i)$ is tested against the threshold before scoring. Its CVR_{ij} must satisfy τ_j, *i.e.,* the inequality $CVR_{ij} \geq \tau_j$ must be satisfied. Notice that not all scores carry monetary value. To have a uniform pricing policy across all treatments we are using first price auction ($p_{ij} = b_j$).

Experimental Results and Observations

As described earlier in Section 5.3.3 we had only limited visibility into performance of the treatments. In particular, we had number of impressions, number of completed views and number of conversions on per (treatment, ad space, ad) tuples. To get a better understanding about the impact of the experiment to both the platform as well as the demand side (or ads) we consider results at two levels. At the platform level we want to make sure that the total revenue is high, however at ad campaign level we want to see that ads j have good ω_j's.

Platform Level

To analyze a market's key performance indicators (KPI's) we calculate and consider statistics defined in Table 5.1.

At the platform level, we have observed the following. *eCPM – control* generated the highest revenue and the lowest VPS. This indicates that clicks of the ad do not necessarily correlate with user intention to convert, given the click view, *i.e.,* CTR_{ij}'s

Metric	Description
$\#impressions$	Total number of impressions served by the treatment. Notice that this metric is distinct from the number of *requests* served by the treatment; it only accounts for impressions served.
$\#cv$	Total number of completed views.
$\$revenue$	Total amount of payments made by the ads for impressions served through the treatment (in U.S. dollars).
$\#installs$	Total number of installations attributed to the treatment.
CR	Rate of completed views in the treatment. $CR = \frac{\#cv}{\#impressions}$.
$PVCR$	Rate of post view conversions (installs) in the treatment. $PVCR = \frac{\#installs}{\#cv}$.
$eCPM$	Average revenue per 1000 impressions to the publisher. $eCPM = \frac{\$revenue}{0.001 \times \#impressions}$.
VAL	Total value received thought installs attributed to the ad or group. We calculate it using CPA targets w_j's total value to ad j val_j is $val_j = \#installs \times w_j$. Total value to the treatment is $val_{\mathcal{J}} = \sum_{j \in \mathcal{J}} val_j$.
VPS	This metric calculates the value per spend $VPS = \sum_{j \in \mathcal{J}} \frac{val_j}{\$spend}$. It indicates the average efficiency of the dollar spend by the advertiser.

Table 5.1: List of metrics used for experiment evaluation.

and CVR_{ij}'s are uncorrelated; we have confirmed this observation. In the platform this can in turn be explained by the types of impressions present in the market. For instance, some publishers are requesting video ads and have *non skippable autoplay* ad slots, meaning that once the user reaches the place in the app with the ad, the ad starts playing automatically, and the user cannot skip it, guaranteeing *completed view*. Another type of inventory that can further support the observation is *rewarded*. In rewarded ad slots the user is encouraged to click on the ads for in-app bonuses.

Next we consider the performance of *Linear, Daily Thresholds*. The experimentation framework gave all the treatments a comparable number of ad requests. We observe, that *Linear, Daily Thresholds* have served 49% fewer impressions and 53% less revenue ($\$revenue$). Upon investigation of CVR_{ij}'s and τ_j's, we have found that threshold values τ_j's appeared to be too high in practice. This situation could have happened due to a number of factors: threshold τ_j's update function is suboptimal, CVR_{ij}'s have low quality, or possibly target w_j's are not realistic and are impossible to satisfy.

Linear collected only 2.5% less revenue than *eCPM – control*. Its KPI's are among the best across the board: VAL, VPS. Note that this group delivered the largest number of clicks and had the highest post click conversion rate. Looking further into the data we have discovered that *Linear* allocates more impressions to ads with relatively low w_j's. There are ads with w_j =\$20, and there are ads with w_j = \$3. Interestingly, both predicted and empirical conversion rates for the latter are higher, telling us that these conversions are easier to find, and hence advertisers are willing to pay less.

We observe that *Credit CPA* produces the highest value Val and VPS, while its absolute number of installs is one of the lowest, followed only by *Threshold –daily.* We also observe that it delivers fewer, but with higher, corresponding w_j's conversions. *Credit CPA* favors ads j with higher target w_j's because (i.) formulation of the scoring function ($\lim_{w_j \to \infty} s_j^{credit} = 2CTR_{ij}b_j$); (ii.) the ads with higher w_j's more often get a positive credit, because inequality $\frac{\alpha_{ij}}{w_j} \leq CVR_{ij}$ is easier to satisfy. (iii.) We observe that formulation of *Credit CPA* is too harsh when computing score for impressions with lower CVR_{ij}'s, because the formulation assumes that we are going to make up for the deficit by buying impressions of a similar quality.

Ad Campaign Level Evaluation

At the ad campaign level, it is important how ad campaigns $\mathcal{J}$ are performing under different allocation methods: how many ad campaigns meet their target w_j's. Since we haven't found an allocation rule that guaranteed to deliver w_j's to ads, we are interested in finding an allocation rule that improves on the empirical conversion rates ω_j's. In addition, one can measure the efficiency of the platform by measuring the fraction of the money that was spent on the platform in an efficient way, *i.e.,* s.t. w_j's are satisfied or improved. In what follows, we consider the impact of treatments at ad campaign levels. We compare each treatment group to the control group (*eCPM - control*). We consider a number of metrics, such as: (a.) number of distinct ad campaigns that received impressions; (b.) amount of money spent by Treatment vs Control; (c.) value delivered by Treatment vs Control; (d.) VPS delivered by Treatment vs Control. For each control and treatment pair we partition data into 11 categories, based on the following criteria:

- **meets w_j in control:** denoted by $C+$ if ad campaigns meet their target w_j's, and $C-$ otherwise;

- **meets w_j in treatment:** denoted by $T+$ and $T-$, respectively;
- **improved:** let ω_j^{tr} and ω_j^c be empirical cost of acquisition in treatment and control groups, respectively. Therefore we can say that ad j was *improved*, if $\omega_j^{tr} \leq \omega_j^c$. It takes a binary value.

Hence partitions can be defined by three dimensional tuples $(C\{+,-\}, T\{+,-\}, \{T, F\})$. For instance, partition $(C+, T+, F)$ contains observations for ad campaigns that were meeting w_j's in both control and treatment, but had $\omega_j^{tr} > \omega_j^c$. For some ads we did not observe installs within a treatment (or control). We denote this event by 0. For instance, partition $(C-, 0, F)$ contains observations for ad campaigns that were not meeting w_j's in control, and for which we did not observe installs in the treatment. In these cases *improved* is set to $FALSE$.

Two out of three treatments showed positive results. First, *Linear* allowed 20% of all ad campaigns to meet their target w_j's and has improved ($\omega_j^{tr} \leq \omega_j^c$) 30%. In 6% of cases it took ad campaigns from $C-$ to $T+$; in 15% of cases it improved ω_j^{tr}, though not enough to allow ads to meet their targets. Most importantly, *Linear* delivered conversions and w_j's to 7% of ad campaigns (*i.e.,* in control group these ads had *no* conversions). We have also observed that *Linear* has spent insignificantly less money (~1.5%) in partitions in which it has delivered higher empirical conversion prices. It has delivered by far more installs (30%), and as a result, ads received more value and higher VPS in the treatment.

The second stand out was *Linear, Daily Thresholds.* The treatment allowed 23% of ad campaigns to meet their w_j's and has improved 37% of all ad campaigns. Also, in the partition $(C-, T-, F)$ the treatment spent 55% less money in comparison to the control. Intuitively, this method is filtering out low quality impressions that the platform classified as "unlikely to converge" by estimating low CVR_{ij}'s. Over all *Linear, Daily Thresholds* have significantly improved value and value-per-spend delivered to ad campaigns. For instance, it has doubled VPS for ad campaigns in partition $(C-, T+, T)$!

Finally, honorable mention goes to *Credit CPA*. It allowed 18% of ad campaigns meet to their targets, and has improved ω_j's for 33%. However, despite the comparable number of impressions and revenue, it has delivered only 0.87 of installs delivered by the control and only 3% more value than control.

5.4 The Advertiser Perspective

Advertisers are strategic players in the market. Advertisers analyze performance of the market, and outcomes achieved by various values of b_j, B_j, w_j, etc. The advertiser may decide to modify any of the campaign settings. They will do so if they find that it is beneficial for them. For instance, by lowering the bid they will be winning fewer impressions, this will also affect the empirical CPA's.

To understand the advertiser perspective better we introduce the following dynamic programming. It helps to compute the best-response strategy for a single advertiser. Let set of bid(s), budget(s) and CPA(s) be the strategy of the advertiser. Then we define the *best-response strategy* of the advertiser as a strategy, that produces the best favorable outcome for the advertiser taking that other advertisers' strategies are fixed.

Optimal Set of Impressions for a Fixed Ad

The main disadvantage of the Knapsack based method introduced in Section 5.3.2 is that in some cases it will not allow advertiser to participate in allocation of impressions if $\frac{CVR_{ij}}{CPC_{ij}}$ falls below threshold τ_j. However, it can be beneficial for the advertiser to get a fraction of impressions with very high conversion-to-cost ratio and a number of impressions with very low conversion-to-cost ratio. To find the optimal allocation for the fixed advertiser j disrespect to the ratio consider the following dynamic programming approach.

Consider allocation problem from a perspective of a fixed advertiser j. For convenience of notation we omit index j in this section. Let $S(a, l, k)$ be the maximum $\sum_{i \le a} x_i CTR_i$ after processing impression a such that

$$\sum_{i \le a} x_i CTR_i CPC_i \le l \quad \text{and} \quad \sum_{i \le a} x_i CTR_i CVR_i \le k$$

Then,

$$
\begin{aligned}
S(a,l,k) = \max\{ & \\
& S(a-1,l,k), && \text{don't allocate} \\
& S(a-1,l-CTR_aCPC_a,k-CTR_aCVR_a)+CTR_a && \text{allocate} \\
& \}
\end{aligned}
$$

We compute $S(a,l,k)$ for all $a \in I$, $l \leq B$ and $k \leq I$, where I is the maximum number of installs for advertiser j. To consider all values for l and k we iterate over corresponding intervals with the step of κ. Thus, the number of S's we estimate is,

$$|I| \times \frac{B}{\kappa} \times \frac{|I|}{\kappa}$$

and each takes $O(1)$ time to compute, and hence, the overall running time is

$$O\left(\frac{|I|^2B}{\kappa^2}\right)$$

which is pseudopolynomial in input size due to dependence on B.

Computing Best-Response Strategy for Classes of Impressions. Ad networks serve billions of impressions in a single day. For instance, in 2015 Google served 30 billion impressions daily[3]. It is not realistic to expect estimates on per impression basis. To produce estimates, platforms frequently cluster impressions into classes, and produce estimates per class. It is more realistic to assume, that platform is given a *landscape* of impressions. The landscape consists of the set of classes S, each class $s \in S$ has the following estimates: (i.) number of expected impressions: n_s; (ii.) number of expected clicks: C_s; (iii.) number of expected conversions: A_s; (iv.) total expected spend: P_s. Assume that advertiser j bids probabilistically on each class, that is, bidding π_s on class s gets π_s of each of these quantities.

Define $T(a,\alpha,\beta)$ to be the solution with maximum clicks with total expected number of conversions $\leq \alpha$ and total expected spending $\leq \beta$, using class $[1,i]$ (assume classes are numbers $1,...,|S|$). Then,

---

[3]`http://venturebeat.com/2012/10/25/30-billion-times-a-day-google-runs-an-ad-13-million-times-it-works/`

$$T(i, \alpha, \beta) = \max_{\pi}\{T(i-1, \alpha - \pi A_i, \beta - \pi P_i) + \pi C_i\}$$

Say π's are in multiples of κ_1; expected conversions are in multiples of κ_2 and expected spend is in multiples of κ_3. Then the complexity of the dynamic programming above is:

$$O((1/\kappa_1)|S|(I/\kappa_2)(B/\kappa_3))$$

where I is the total number of impressions, and B is the maximum expected spend. This solution will be approximate, but is the best solution among those where α and β are rounded to $|S|\kappa_1\kappa_2$ and $|S|\kappa_3\kappa_1$, respectively.

5.5 Related Work

Motivated by online ad markets, several variations of this problem have been studied. Some, that received most of the attention are presented below.

Adwords – Matching Ads to Search Queries Adwords [39] problem is the simplification of ad allocation problem that arises in ad networks when matching search queries of users to advertisers. The platform has a set of advertisers $\mathcal{J}$. Each advertiser has a daily budget B_j and a bid b_j. When an impression request arrives, the set of eligible ad campaigns is revealed, which then must be allocated to one of theeligible ad campaigns. An ad campaign, when allocated, is charged b_j. The ad campaign cannot participate in allocation once it exhausts its daily budget B_j. The objective is to maximize the total amount of money spent by advertisers. The problem is a generalization of online bipartite matching, in which each offline node ($j \in \mathcal{J}$) has a budget B_j associated with it. More formally,

$$\begin{aligned}
\text{maximize} \quad & \sum_{ij} x_{ij} b_j \\
\text{subject to} \quad & \sum_{j} x_{ij} b_j < B_j \ \forall j \\
& \sum_{i} x_{ij} \leq 1 \qquad \forall i \\
& x_{ij} \in \{0, 1\} \qquad \forall i, j
\end{aligned} \tag{5.9}$$

where x_{ij} is an indicator variable that is set to 1 if ad j was allocated impression i, and to 0 otherwise.

Display Ads – Matching display ads to impressions. The display ads problem was introduced in [19], being inspired by the problem of matching display ads to users' impressions which arrive online. In this market the contracts between platform and advertiser usually specify the total number of impressions. Hence, the description of the problem is as follows. The platform has the set of advertisers $\mathcal{J}$. Each advertiser j has a capacity c_j, that indicates how many times the ad can be matched to incoming impressions. When an impression arrives, it can be matched to a subset of eligible ads $\mathcal{N}(i)$. For each impression i and ad j pair, there is a quality factor w_{ij} that represents how well the ad j matches for impression i. The goal is to maximize total weight of vertices matched while respecting capacities of advertisers c_j's. More formally,

$$\begin{aligned} \text{maximize} \quad & \sum_{ij} x_{ij} w_{ij} \\ \text{subject to} \quad & \sum_{j} x_{ij} < c_j \ \forall j \\ & \sum_{i} x_{ij} \leq 1 \quad \forall i \\ & x_{ij} \in \{0,1\} \quad \forall i,j \end{aligned} \tag{5.10}$$

There is a Generalized Assignment Problem (GAP), that generalizes both of the above problems. Here w_{ij} is the weight of the edge, and c_{ij} is the cost.

$$\begin{aligned} \text{maximize} \quad & \sum_{ij} x_{ij} w_{ij} \\ \text{subject to} \quad & \sum_{j} x_{ij} c_{ij} < B_j \ \forall j \\ & \sum_{i} x_{ij} \leq 1 \qquad \forall i \\ & x_{ij} \in \{0,1\} \qquad \forall i,j \end{aligned} \tag{5.11}$$

Optimization of budget-constrained advertisers. The above problems tackle objective functions that optimize an objective of the ad network allocating impressions to ads. However, the ad network is an intermediary between publishers and advertisers. Therefore, the ad network should strive to optimize the experience of both sides of the market, rather then its own. Ad network can choose to optimize the number of statistics for each of the sides, for instance, revenue, quality, ROI, *etc.*

In [28] authors develop solutions that in expectation, based on estimated probabilities CTR_{ij}'s finds an allocation rule that optimizes the variety of objectives for advertisers. Their work solves the allocation problem with expected costs of conversion, *e.g.,* optimizes expected cost for conversion for budget-constrained advertisers. Note that ω_j can differ from expected cost of conversion $\sum_{i \in \mathcal{N}(j)} x_{ij} CTR_{ij} CVR_{ij} w_j$, where $x_{ij} = \{0, 1\}$ is an indicator variable. In our case we are interested in finding an allocation rule that gives an allocation that is (1.) revenue effective from the platform perspective (e.g., comparable to the revenue of industry standard, GSP); (2.) feasible.

Bicriteria optimization. In [31] offline nodes (ads $\mathcal{J}$) have capacity constraints. The set of online nodes (impressions $\mathcal{I}$) arrives one by one. Upon arrival of online node i, the set of its neighbors $\mathcal{N}(i)$ in offline nodes is revealed, together with its weight w_{ia} for eligible neighbor j. The problem is to assign each arriving item i to one of its eligible neighbors, or discard it, while respecting constraints. The goal is to maximize the cardinality of the matching (number of impressions) and the total weight of the matching.

In [3] authors consider the variation, which is a strict generalization of the standard setting. There are offline nodes $\mathcal{J}$. The set of online nodes $i \in \mathcal{I}$ arrives one at a time. Upon arrival of a node i a set of adjacent offline nodes $\mathcal{N}(i)$ is revealed. The edges connecting the nodes can be of two types (schematically): *red* and *blue*. The goal is to find a matching, which contains a large number of edges of each color. Projected to an ad allocation scenario, one can consider one set of edges (*e.g.,* red) beneficial for one side of the market (*e.g.,* advertisers, ads), and the other set of edges (*e.g.,* blue) for the other (*e.g.,* publishers).

5.6 Discussions

User Lifetime Value and Its Role. In display online advertising the majority of advertisements are promoting a particular product or service. The value acquired per conversion is frequently assigned to the conversion event such as a *"purchase of Nike running shoes."* Apps are increasingly offered for free[4], but monetized through advertising or in-app purchases of subscriptions or virtual goods.

- In-app advertising — an app creator's goal is to create an app with interesting content, maintain a large and stable userbase and collect user-related information to later sell to other brands and app publishers, who pay to place targeted ads in the app (for instance, NBA Game Time[5]).

- Freemium — an app creator launches an app that can be installed for free, however, certain "advance" or "premium" features are gated and cost money to be unlocked. Many successful gaming apps fall into this category; they have found success by making basic game versions free to all users, but requiring the user to purchase additional levels or premium options. AngryBirds[6] is one such example; it has a basic version with ads, and a premium version without ads. The premium version has significantly more levels.

- In-app purchases — Apps are sales channels, and show the user listings of products. Note that an app can be selling real world goods (*e.g.,* sporting gear, furniture, handmade jewelry) or *virtual goods* (*e.g.,* an imaginary shovel, which allows the user to plant an unusual crops and earn more game points). Farmville[7] makes their money from users completing offers to receive bonus cash for their in-game play.

- Sponsorships, also known as *incentivized advertising*, is the newest form of monetization strategy. The app is monetized through partnering with advertisers, who provide *promotions* or *rewards* for certain actions. For instance, RunKeeper[8] motivates users to track their running activities by offering rewards upon reaching

[4]`http://flurrymobile.tumblr.com/post/115189750715/the-history-of-app-pricing-and-why-most-apps-are`

[5]`https://itunes.apple.com/us/app/nba-game-time-2014-15/id335744614?mt=8`

[6]`https://www.angrybirds.com/`

[7]`https://zynga.com/games/farmville-2`

[8]`http://runkeeper.com/`

certain levels. The promotions are sponsored by advertisers, and the app earns revenue by taking a share of the revenue generated to the advertiser.

Intuitively, app developers are monetizing their users, and their revenue consists of the net profits generated by distinct users. The amount of money the user spends in the app depends on a number of factors, most importantly the app itself and the user. For instance, a young female who lives in New York city and uses a shopping app generates more revenue for the app (via in-app purchases) than a middle-aged man who lives in Mountain View, CA uses NBA official app to follow the tournament (via viewing in-app ads). Of course app developers want to expand their userbase and increase their revenue. However, to stay net positive, app developers need to acquire new users at the price below the revenue generated by these users. In particular, app developers track their users to estimate the users' *life time value* (LTV), where LTV is a prediction of the net profit attributed to the entire future relationship with a customer. Due to the sizes of audiences and data sparsity, LTV's are frequently produced not on a per customer basis but per *market segment* (group of users who has an attribute in common). Advertisers are creating sophisticated targeting strategies (Section 2.1), which allow them to reach their target audiences at an acceptable price.

One possible way to calculate target w_j's as discussed in Section 5.1, or maximum price per conversion advertiser of j is willing to pay is as follows:

$$w_j = \frac{1}{|Users|} \sum_{u \in Users} (LTV_u) - cost - profit$$

where LTV_u is expected lifetime value of user u, $cost$ is the development cost, and $profit$ is the revenue objective. Therefore, w_j is ad campaign j specific, and represents the average cost-per-user acquisition that advertiser j can spend and still earn the revenue objective. We refer to w_j as *cost-per-acquisition target*, or *CPA target*.

Trade Off of Empirical CPA vs. Budget. Clearly, there is a trade off between number of impressions and clicks and number of conversions that ad j receives. For instance, consider ad j, that has $w_j = \$2.00$ and bid $b_j = \$0.20$. Say the ad network, when serving the ad, can choose between two outcomes:

1. deliver 100 impressions, 2 clicks and 1 conversion at $\omega_j^1 = \$0.40$. ω_j^1 satisfies w_j, hence the requirements of j are satisfied;

2. deliver 1000 impressions, 30 clicks and 4 conversions at $\omega_j^2 = \$1.50$. Again, ω_j^2 satisfies w_j, hence the requirements of j are satisfied.

While in both cases ω_j of ad j is satisfied, we believe that from the advertiser's perspective the second outcome is preferred. Because in the first outcome, ad j total value $VAL^1 = \$2.00 \times 1 = \$2.00$, while in the second outcome total value $VAL^2 = \$2.00 \times 4 = \$8.00$. In the first case, the advertiser gets users at a more favorable rate, however she acquires more value by acquiring more users, despite the fact that $\omega_j^1 < \omega_j^2$. Therefore we refer to w_j as *as targets* and not thresholds.

5.7 Conclusion

We introduce a new class of allocation problems with secondary targets. We discuss number of theoretical approaches to the problem. We test Knapsack and allocation based approaches on live traffic of Flurry ad network and observe positive results in terms of revenue and number of conversions. As the result, the platform has switched to use the *Linear* allocation rule. While it does not have any theoretical guarantees, it performs well in practice. It has been shown to generate revenue, comparable to *eCPM – Control*, yet delivers significantly more installs and hence greater value to advertisers. The *Linear, Daily Thresholds* method has shown itself to be a promising direction for the solution of delivering $\omega_j \leq vj$ to most ad campaigns. However, the implementation tests resulted in a significant cut in impressions delivered and revenue obtained. More sophisticated threshold update function has to be developed prior to platform-wide deployment of the approach.

Each of suggested methods can be developed into a theoretical framework.

Chapter 6

Future Directions

Markets developed headlong. A number of different markets emerged, each with unique properties and challenges. They inspired numerous problems in research (*e.g.,* progress in online allocation) and pushed developments in systems. Indeed, the ad for an impression needs to be chosen and served in less than 200 ms and it has to be done billions of times per day.

The role of information is crucial as markets become smarter and more elegant. More and more information is flown into the markets. However, there is a lack of understanding about which information will improve the efficiency of an ad campaign, and which aspects of user profiles affect user experiences. We introduce adscape, an advertising landscape study of display and video ads. In the study we have fixed a number of dimensions, including time. However, it is crucial to understand the role of time in the adscape. Observation of the adscape is very resource intense. It would be beneficial to develop generative models for adscapes. Finally, what is the impact of discovered information on strategic players? [1] models information as *good* or *bad*, their model does not account for other aspects of user profile, such as interests.

Further, markets develop and become even more complex and sophisticated. For instance, consider market of search ads, where the platform can choose to show some combination of text and product ads. What is the optimal auction to run? What are the properties of this auction? Further, consider goals of advertisers in the market. Today, they are bidding and paying per click. However, intuitively their goal is not a click, but a conversion. Advertisers have *secondary targets.* What is an ad allocation problem for search ads in the presence of various types of advertisers with secondary targets?

Finally, the rise of markets did not stop at advertising. The market approach spread and opened up new directions. For instance, today there are shared economy platforms for taxis, apartments and boats. What unique problems have arisen on these markets? What unique markets will arise in the future?

References

[1] I. Abraham, S. Athey, M. Babaioff, and M. Grubb. Peaches, lemons, and cookies: designing auction markets with dispersed information. In *Proceedings of the fourteenth ACM conference on Electronic commerce*, pages 7–8. ACM, 2013.

[2] Z. Abrams, S. S. Keerthi, O. Mendelevitch, and J. A. Tomlin. Ad delivery with budgeted advertisers: a comprehensive LP approach. *Journal of Electronic Commerce Research*, 9(1):16–32, 2008.

[3] G. Aggarwal, Y. Cai, A. Mehta, and G. Pierrakos. Biobjective online bipartite matching. In *Web and Internet Economics*, pages 218–231. Springer, 2014.

[4] G. Aggarwal, J. Feldman, and S. Muthukrishnan. Bidding to the top: VCG and equilibria of position-based auctions. *Approximation and Online Algorithms*, pages 15–28, 2007.

[5] G. Aggarwal, A. Goel, and R. Motwani. Truthful auctions for pricing search keywords. In *Proceedings of the 7th ACM Conference on Electronic Commerce*, pages 1–7. ACM, 2006.

[6] A. Arasu and H. Garcia-Molina. Extracting Structured Data from Web Pages. In *In Proceedings of ACM SIGCOM*. ACM, June 2003.

[7] S. R. Balseiro, J. Feldman, V. Mirrokni, and S. Muthukrishnan. Yield optimization of display advertising with ad exchange. *Management Science*, 60(12):2886–2907, 2014.

[8] P. Barford, I. Canadi, D. Krushevskaja, Q. Ma, and S. Muthukrishnan. Adscape: harvesting and analyzing online display ads. In *Proceedings of the 23rd international conference on World wide web*, pages 597–608. International World Wide Web Conferences Steering Committee, 2014.

[9] A. Broder, M. Najork, and J. Wiener. Efficient URL Caching for World Wide Web Crawling. In *In Proceedings of the World Wide Web Conference (WWW)*, May 2003.

[10] M. Cary, A. Das, B. Edelman, I. Giotis, K. Heimerl, A. R. Karlin, C. Mathieu, and M. Schwarz. Greedy bidding strategies for keyword auctions. In *Proceedings of the 8th ACM conference on Electronic commerce*, pages 262–271. ACM, 2007.

[11] C. Castelluccia, M. Kaafar, and M. Tran. Betrayed by Your Ads! In *In Privacy Enhancing Technologies*, pages 1–17. Springer, 2012.

[12] C. Change, M. Kayed, M. Girgis, and K. Shaalan. A Survey of Web Information Extraction Systems. *In IEEE Transactions on Knowledge and Data Engineering*, 18(10), October 2006.

[13] E. H. Clarke. Multipart pricing of public goods. *Public Choice*, 11(1):17–33, 1971.

[14] B. Edelman, M. Ostrovsky, and M. Schwarz. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *American Economic Review*, 97(1):242–259, 2007.

[15] B. Edelman and M. Schwarz. Internet advertising and optimal auction design. pages 1–1, 2008.

[16] D. Eichmann. The RBSE Spider-balancing Effective Search Against Web Load. In *Proceedings of the 1st international conference on World wide web*, pages 113–120. Geneva, 1994.

[17] E. Even Dar, V. S. Mirrokni, S. Muthukrishnan, Y. Mansour, and U. Nadav. Bid optimization for broad match ad auctions. In *Proceedings of the 18th international conference on World wide web*, pages 231–240. ACM, 2009.

[18] J. Feldman, N. Korula, V. Mirrokni, S. Muthukrishnan, and M. Pál. Online ad assignment with free disposal. In *Internet and network economics*, pages 374–385. Springer, 2009.

[19] J. Feldman, A. Mehta, V. Mirrokni, and S. Muthukrishnan. Online stochastic matching: Beating 1-1/e. In *Foundations of Computer Science, 2009. FOCS'09. 50th Annual IEEE Symposium on*, pages 117–126. IEEE, 2009.

[20] J. Feldman and S. Muthukrishnan. Algorithmic methods for sponsored search advertising. In *Performance Modeling and Engineering*, pages 91–122. Springer, 2008.

[21] A. Ghosh, P. McAfee, K. Papineni, and S. Vassilvitskii. Bidding for representative allocations for display advertising. In *Internet and Network Economics*, pages 208–219. Springer, 2009.

[22] A. Ghosh and A. Sayedi. Expressive auctions for externalities in online advertising. In *Proceedings of the 19th international conference on World wide web*, pages 371–380. ACM, 2010.

[23] P. Gill, V. Erramilli, A. Chaintreau, B. Krishnamurthy, K. Papagiannaki, and P. Rodriguez. Follow the money: Understanding economics of online aggregation and advertising. In *Proceedings of the 2013 Conference on Internet Measurement Conference*, IMC '13, pages 141–148, New York, NY, USA, 2013. ACM.

[24] T. Groves. Incentives in teams. *Econometrica: Journal of the Econometric Society*, pages 617–631, 1973.

[25] S. Guha, B. Cheng, and P. Francis. Challenges in measuring online advertising systems. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pages 81–87. ACM, 2010.

[26] M. Gupte, D. Krushevskaja, and S. Muthukrishnan. Analyses of cardinal auctions. *Algorithmica*, 71(4):889–903, 2015.

[27] B. Kalyanasundaram and K. R. Pruhs. An optimal deterministic algorithm for online b-matching. *Theoretical Computer Science*, 233(1):319–325, 2000.

[28] C. Karande, A. Mehta, and R. Srikant. Optimizing budget constrained spend in search advertising. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 697–706. ACM, 2013.

[29] R. M. Karp, U. V. Vazirani, and V. V. Vazirani. An optimal algorithm for on-line bipartite matching. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 352–358. ACM, 1990.

[30] T. Kesselheim, K. Radke, A. Tönnis, and B. Vöcking. An optimal online algorithm for weighted bipartite matching and extensions to combinatorial auctions. In *Algorithms–ESA 2013*, pages 589–600. Springer, 2013.

[31] N. Korula, V. S. Mirrokni, and M. Zadimoghaddam. Bicriteria online matching: Maximizing weight and cardinality. In *Web and Internet Economics*, pages 305–318. Springer, 2013.

[32] D. Krushevskaja and S. Muthukrishnan. Market approach to social ads: The mylikes example and related problems. In *Algorithms and Computation*, pages 1–10. Springer, 2013.

[33] R. P. Leme and E. Tardos. Pure and Bayes-Nash price of anarchy for generalized second price auction. In *Proceedings of the 2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 735–744. IEEE Computer Society, 2010.

[34] B. Liu, A. Sheth, U. Weinsberg, J. Chandrashekar, and R. Govindan. AdReveal: improving transparency into online targeted advertising. In *Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks*, pages 1–7. ACM, 2013.

[35] B. Lucier, R. Paes Leme, and É. Tardos. On revenue in the generalized second price auction. In *Proceedings of the 21st international conference on World Wide Web*, pages 361–370. ACM, 2012.

[36] S. Martello and P. Toth. *Knapsack problems: algorithms and computer implementations*. John Wiley & Sons, Inc., 1990.

[37] O. McBryan. GENVL and WWWW: Tools for Taming the Web. In *Proceedings of the 1st international conference on World Wide Web*, pages 79–90. Geneva, 1994.

[38] A. Mehta. Online matching and adwords. In *Approximation Algorithms Workshop, Princeton, USA*, 2011.

[39] A. Mehta, A. Saberi, U. Vazirani, and V. Vazirani. Adwords and generalized online matching. *Journal of the ACM (JACM)*, 54(5):22, 2007.

[40] V. Mirrokni. Online ad allocation, and online submodular welfare maximization. 2012.

[41] S. Muthukrishnan. Bidding on configurations in Internet ad auctions. *Computing and Combinatorics*, pages 1–6, 2009.

[42] R. B. Myerson. Optimal auction design. *Mathematics of operations research*, 6(1):58–73, 1981.

[43] C. Olsten and M. Najork. Web Crawling. *In Foundations and Trends in Information Retrieval*, 4(3), 2010.

[44] S. Pandey and C. Olston. User-centric web crawling. In *Proceedings of the 14th international conference on World Wide Web*, pages 401–411. ACM, 2005.

[45] C. H. Papadimitriou and M. Yannakakis. Multiobjective query optimization. In *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 52–59. ACM, 2001.

[46] B. Pinkerton. Finding what people want: Experiences with the webcrawler. In *Proceedings of the Second International World Wide Web Conference*, volume 94, pages 17–20. Chicago, 1994.

[47] D. E. Rivera, M. Morari, and S. Skogestad. Internal model control: PID controller design. *Industrial & engineering chemistry process design and development*, 25(1):252–265, 1986.

[48] H. R. Varian. Position auctions. *International Journal of Industrial Organization*, 25(6):1163–1178, 2007.

[49] W. Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance*, 16(1):8–37, 1961.

[50] J. Yan, N. Liu, G. Wang, W. Zhang, Y. Jiang, and Z. Chen. How much can behavioral targeting help online advertising? In *Proceedings of the 18th international conference on World Wide Web*, pages 261–270. ACM, 2009.