TEL AVIV UNIVERSITY אוניברסיטת תל-אביב
Raymond and Beverly Sackler Faculty of Exact Sciences
Blavatnik School of Computer Science

The Adx Game

A thesis submitted in partial fulfilment
of the requirements for the degree of
Master of Science
in the school of computer science, Tel-Aviv university

by
Tomer Greenwald

The research for this thesis has
been carried out under the supervision of
Prof. Yishay Mansour

February 2018

Contents

Acknowledgments

First and foremost, I would like to thank my family. I thank my parents, Iris and Mendy, for their endless support and care. I thank my sisters, Liat and Meital, and brothers in law, Guy and Joni, for helping and supporting me during my studies.

I would like to give my deepest thanks to my advisor, Prof. Yishay Mansour, and to Dr. Mariano Schain. Through them I was introduced to the fascinating fields of Machine Learning and Ad-Excahge Markets. I thank them for giving me the opportunity to implement their vision and create an online Ad-Exchange competition.

Finally, I would like to thank the members of the TAC-ADX community and to the researchers and students who used this platform. Their help, support, and research contributed and advanced my work and made this project fruitful.

Abstract

Online advertising currently holds the largest market share among advertising platforms. The current most prevalent way to perform advertising content exchange is the Ad-Exchange. While the model is widely researched there has not been a convenient and sufficient platform where researches can test their assumptions and algorithms regarding this trade. The TAC platform was used in the last 15 years as a gateway for researchers in the fields of artificial intelligence and multi-agents systems to empirically compare approaches for trade related problems. We developed a model which incorporates the Ad-Exchange model with the TAC platform. This model allowed multiple teams from universities around the world to build trading agents and asses their performance in the Ad-Exchange domain.

Chapter 1

Introduction

1.1 AdX model

Online advertising currently holds the largest market share among advertising platforms (such as cable/brodcast TV, Radio, etc.). This platform encompasses web sites (both mobile and desktop), mobile applications and all other digital means for content delivery over the Internet platform. Sponsored search ads and Display ads (banners and videos) are the main tools used in this platform. Total spending for display advertising in the US has suppressed 70 billion US$ last year. More than 50% of the total revenue share is held by two companies: Google and Facebook, who control the two largest advertisement exchange platforms.

Display ads are used by merchants and organizations to sell and promote merchandise or to advance ideas and engage with customers. Facebook, the largest social network in the world and the second largest display advertising market share holder, generates more than 95% of its revenue through its display ads platform. More than 200 billion US$ were invested in the display ad market last year.

Ad-Exchanges were not always around. At first, publishers who wanted to gain revenue from available display space on their sites sold it directly to advertisers. Since not all available space was sold, publishers contacted Ad Networks to help them with the process of selling the remaining product. These Ad Networks offered their clients ad spaces that they obtained from the publishers at a discounted price since they bought it in masses. Contacting multiple ad networks could be considered as a hassle for the publishers and so the next evolutionary step was the introduction of network optimizers which offered a smoother experience for both ad networks and publishers with regard to which ad network should serve which publisher at any given time. The last step in the development process of online advertising came in the form of the Ad-Exchanges. These exchanges allowed direct audience targeting for advertisers and offered a single contact point for ad serving to publishers. This allowed publishers to serve ads faster than before and greatly increase the monetization value of their available ad space.

The current most prevalent way to perform this exchange is the Ad-Exchange model. Google and Facebook's exchanges are built in a similar fashion. The model's flow starts with a user browsing a website, or using an application. The webstie contacts the exchange and offers one, or more, slot(s) to be sold to interested parties. The Ad-Exchange sends details about the user and the website, or application, to advertising networks, Ad-Networks, that have shown interest in groups that this user is classified in. Each interested Ad-Network sends a bid to participate in an auction for the available slots in a Generalized Second Price (GSP) auction, that is held to determine the winners who will get their ads shown to the user and the price paid for them.

1.2 TAC model

Over the last 20 years multiple approaches were developed to address the different aspects of online commerce. Applied research in artificial intelligence in general, and multi-agents systems in particular, was targeting this important domain. The high level goal has been to empirically compare various methodologies, and develop a coherent understanding of them. The Trading Agent Competition (TAC) [15] offered a standardized model which allowed teams from around the world to develop their own agent mechanisms and compete against one another in a simulated e-commerce market. The TAC model attempted to model a real life scenario that involves bidding strategies for related goods. This attempt led to the first *Trading Agents Competition* which was was in held in a workshop at the 2000 ICMAS conference with 20 competing teams from around the world.

The simulated game offered by [15] models travel agents and their attempt to arrange trip itineraries for their clients. Each agent is assigned throughout a *game* with tasks that involve requests from clients who wish to travel. They need to arrange for each client a complete travel itinerary, including booking a flight, reserving a hotel for the desired time period, and register for entertainment events. The agent is responsible for purchasing these items in accordance to the demand presented by the customer. The agent's utility function is based on the successful allocation of the requested items, completing specific requests made by the customer and the expenditure it made in the procurement process.

Different games of the *TAC* soon after surfaced, each addressing a different commerce aspect. The Supply Chain Model (*TAC-SCM*), offered by [1], modeled an assembly supply chain which involves manufacturers, component suppliers, and customers. When a request for a quote was allocated to a manufacturer it was responsible for the assembly process. Multiple parts were to be procured, while some of them were relied on others being compatible with their type. A different approach to the subject was offered by [9] called CAT (reverse of TAC). In this model the buyers and sellers were introduced by the game designers. The participants needed to implement exchange market specialists that matched the buyers and sellers trade offers. The specialists made profit by charging traders on various services provided to them, such as: daily registration fee, placing a trade offer, obtaining information about the market state and so on.

The fourth simulated game in the TAC series introduced the advertising world. The TAC Ad Auctions,TAC-AA, [6], focused on sponsored search advertisement and the interaction between ad networks, publishers and users. The TAC-AA modeled an environment where players assumed the roles of an advertiser and needed to compete with other advertisers on user impressions and expenditures following an online search for products. Advertisers bid on sponsored search ads. During the game users were querying for products, being presented by sponsored ad impressions, clicking ads and eventually buying, or not, the commerce products. Advertisers were responsible for keeping a model of users distribution in the world, and to track how many of them converted from state to state in order to gain higher revenue in the game.

The TAC-AA simulation was the first step in the TAC series toward online advertising simulations. A predecessor competition model was the *Pay Per Click Bidding Agent Competition* which took place in ACM EC-06 and allowed participants to manage live campaigns for keywords under the Microsoft AdCenter. Though sponsored search advertising modeling was successful in the TAC world it only modeled sponsored search, and not other online advertising domains.

Brand Advertising, in which effectiveness and purchase intentions are hard to measure does not fit under the TAC-AA model. Furthermore, the existence of an Ad-Exchange which facilitates a large portion of the communication between ad networks and publishers was not modeled in the TAC-AA model that assumed a direct line of communication between advertisers and publishers. This has lead to the development of the TAC-ADX game. The new game, introduced to the world of TAC the concept of *Ad Networks* - agents responsible for advertising campaigns of multiple clients and an *Ad-Exchange* - which is the main and sole coordinator both Ad Networks and publishers who wish to commerce with each other. *Ad Networks*, implemented by competing teams, need to manage multiple, sometimes overlapping, contracts for their customers.

1.3 Research

This research presents a new model for the TAC competition which allows the modeling of brand advertising and researching different advertising strategies. The new model's purpose is to allow research of different components in the brand advertising market with a large set of available agents. Each agent implementing a different approach for campaign management. The model's ability to simulate real world behavior allowed it to act as a research platform for groups in the advertising smart agents domain, [12], [4]. Additionally, this new platform allowed us to research the effects of tempering with the publishers' reserve price selection algorithm. The research shows the relation between reserve price selection algorithms to bid values and total income over time.

In the second chapter the Ad-Exchange model is detailed for its different components along with the research and adaptations made to incorporate it to the TAC platform. The simplified model shows the main agents involved in the online advertising trade market: Users, Publishers, Ad-Networks, and the Ad-Exchange. The specific TAC model presents ideas and services in this domain that relate to the simulation, e.g, User Classification Services and Advertising Campaigns.

The third chapter shows the research over how reserve prices affect publishers' income and bid values. We compared multiple approaches for reserve price selection: (1) Static approach (2) Dynamic approach (3) Approach based on difference of convex functions. The last approach was shown to the most effective across multiple simulations.

The last chapters presents the structure of the platform built for the research - TAC-ADX. The challenges we faced and solutions provided for building a real time competition platform supporting remote agents. The workshops ran in Tel-Aviv University and the international TAC competitions for which this platform was developed.

Chapter 2

AdX - Ad-Exchange

2.1 Simplified Model

When creating a game for the Ad-Exchange domain, we clearly need to give a simplified model. Simulating an Ad-Exchange environment requires first to understand what are our expected goals from this simulation, which ingredients (elements or processes) of the environment are essential, and which are optional. We also need to consider the impact on the complexity of the system that each of them introduces. Each architectural choice we make can either simplify or make our model more complicate. Therefore we start by listing what we except to achieve with the simulation and derive the necessary components for these goals.

One of the first choices made was selecting who is the target audience to use this simulation and what do they expect from it. Since we were familiar with the TAC environment it was natural for us to turn to the TAC community and offer our future model as a TAC competition. Therefore the model needed to be competitive, e.g, have a clear utility function which players were expected to maximize, be built as a single-server multiple-agents platform, and, of course, incorporate key features from the Ad-Exchange world on which the platform will revolve.

We identified key players and their features that encompass with their actions and interactions most of the significant interaction in this domain, which were relevant to our cause. Building a simulated environment for an Ad-Exchange in order to evaluate player performance required a standardized model in which these key features were included. We found the model offered by [8] as an excellent starting point and eventually extended it with [11]. While the basic model lays the foundation for an academic and quantitative approach to the Ad-Exchange and Brand Advertising market, the extended model details a specific implementation. In it details regarding how agents interact with each other, mechanisms to govern resource allocation, and rules to abide by and dictate how agent performance should be measured.

There are four main players in the online advertising exchange market who interact with one another to enable this trade. The *Users* are people who use online platforms. They are targeted by *Ad-Networks* to show them ads for brands and products. The ads are delivered to the *Users* while they interact with the online platforms, e.g, websites and applications, and these will be referred to as the *Publishers*. The *Publishers* monetize the *Users* by contacting an *Ad-Exchange* to facilitate this trade deal. For every ad space impression the *Publisher* will contact the *Ad-Exchange* which in turn auctions among *Ad-Networks* the opportunity to present their ad to the *Users*.

2.1.1 Users

Users are the driving force behind the entire Ad-Exchange market and therefore were a clear candidate as a key player in the model. These are everyday people who browse websites, play on their smart phone, watch videos and use online applications. They are both the target of online advertising and a product in this market. This trait makes them a valuable asset to all other exchange players and gives them their dominant role, both in the real world and in our model.

Users, specifically user generated traffic, are often traded and monetized in the Ad-Exchange market between *Publishers* and *Ad-Networks*. Either by non-targeted bulk impressions sold by publishers at a low price to Ad-Networks, who wish to publish a brand or a product to a great deal of audience, or by specially crafted ads directed at a specific user profile. Users are monetized constantly as a commodity by the exchange market. In order to enrich a commodity's value one must be able to attain as many unique characteristics for it to differentiate between types of goods. In our domain, a user's defining characteristics are those which can help identify it, either specifically as an individual or as a member of an interest group such as gender, male or female, age, young or old, a geographic location, etc. These interest groups are usually the targets for brand advertising which aims to publish its message to a group of people that are likely to buy or use its products. A product can, of course, be sold to multiple groups but sometimes Advertisers would like to direct different messages to each group and this is when the segmentation gains importance.

In the TAC-ADX model several selected attributes are associated with a *user*: *Age*, *Gender*, and *Income level*. Each having multiple categorical values associated with it. Our three categories are some of the widely used characteristics in the real world and their combinations allows the TAC-ADX model to generate enough segmentation in the user population. This includes both overlapping general market segments and unique segments for targeted ads.

2.1.2 Publishers

Our second important key role in the model are the *Publishers*. This group includes websites and any other types of online platforms that monetize their user traffic and available ad space in the Ad-Exchange market. The *Publishers* were in the ad space monetization market long before it became an Ad-Exchange market. Their role is clear and simple - gain revenue from visiting users by placing ads on their platform. The Ad-Exchange is the latest and currently most profitable way to do so and thus the platform evolved from that need and delivers it in better quality than ever before.

Publishers are differentiated from one another by their relative popularity among different segments of users. This user-to-publisher affiliation, e.g, a financial news web site might attract older users than a sports web site, can change the valuation of ad spots to Ad-Networks when they wish to target specific population segments. A popular publisher with a high attraction of users from a specific segment can charge higher prices for its ad slots knowing its value in the market. A secondary, but nonetheless important, characteristic of publishers, which is influenced by the type of population who visits it and the publisher wishes to attract, is the type of ads it shows the users. Ad types can vary from text to banners to video. Each type can generate significantly different revenues for the publishers according to their visiting population and it is the publishers' task to learn which suits them best.

The last selection a publisher makes before an ad can be shown on its platform is perhaps the most important of them all. Long after the ad type it wants to show was selected and all customizations were made with regard to which ads a publisher wants to show on its site, the, often live, auction takes place and the publisher has one variant to play in hopes to shift the results in its favor. This variant is the auction's reserve price which is the minimal price the publisher is willing to sell the auctioned ad space for. In the Ad-Exchange domain the ad auctions are mostly GSP auctions where the reserve price can either invalidate the auction, generate higher revenue, or have no effect on the auction. A thorough review for the subject will be given in the next chapter.

Under the TAC-ADX model we incorporated multiple characteristics for publishers to govern their actions and mimic real world behavior. Among them were a reserve price selection algorithm for which multiple methodologies were given to modeled publishers. This was done to control how reserve prices are selected in a simulation's life-cycle. Relative popularity, the probability of an arbitrary user to visit the site on its daily browsing. User orientation, which denotes how likely a user with given characteristics is to visit the the publisher, e.g, whether a site is oriented towards young-adults, elderly men, etc. Additionally, attributes related to access and ad types, among them how likely a user is to visit the publisher using a certain device type, be it mobile or desktop and how likely an ad of a given type, text or video, is to be shown for a visiting user.

2.1.3 Ad-Networks

Ad-Networks are the gateway, and perhaps the representatives, of the *demand* side in the Ad-Exchange market. These networks are the *buyers* in any given exchange where a *user*'s impression opportunity is sold by a *publihser*. By aggregating ad space inventory, acquired from Publishers/Ad-Exchanges, Ad-Networks can offer their clients, the advertisers, a single point of contact for the purpose of online advertisement by contacting multiple publishers and placing advertisements on their behalf.

Ad-Networks are probably the most competitive players in the Ad-Exchange domain. While users take the back seat when it comes to the actual auctions, publishers can control only auctions related to them and the Ad-Exchange is mostly a monopoly or a duopoly. Ad-Networks can compete in every auction as long as it is relevant to them. An auction's relevance is determined only by an Ad-Network's contracts with advertisers. Therefore, every Ad-Network can, theoretically, compete for every possible contract and they are only limited by their capacity and ability to execute the said contracts in the agreed manner.

To better execute contracts Ad-Networks have multiple tools at their disposal for fine tuning bids towards impressions. A higher or lower bid can determine whether a *good* impression was bought and how much the network needed to spend for it. *Good* impressions, which will be referred to as *targeted impressions*, are those which impress the desired client of a contract signed between and Ad-Network and an advertiser. One of the main tools used for this purpose is a *User Classification Service*. This service allows Ad-Networks to identify *users* according to their relevant *user population*. Whenever a bid is given for a classified impression it is more likely to suite the needs of the advertiser, as opposed to a blind bid.

In the TAC-ADX model we selected the Ad-Networks' role to be implemented and controlled by the competing groups participating in the simulation. Each Ad-Network is required to handle multiple responsibilities given to them. These range from contract signing with advertisers, contract execution by bidding on impression opportunities, handling competition from rival Ad-Networks and many more. Each competing group in the TAC-ADX game designs their Ad-Network agent and attempts to score the highest revenue in the game. More than 20 implementations for Ad-Networks were built by teams from the Tel-Aviv University and from universities around the world, each having different strategies for how to play the role of and Ad-Network in the TAC-ADX game.

2.1.4 Ad-Exchange

The core of the Ad-Exchange domain, both in the real world and in our model, is the *Ad-Exchange* itself. Having the entire domain called by its name hints at its significance. The *Ad-Exchange* acts both as a platform and as in intermediary for both sides of the exchange, the demand, *Ad-Networks*, and the supply, *Publishers*. By offering a unified online exchange platform where *Publishers* offer user impression opportunities and *Ad-Networks* compete for their clients' ads to be shown it draws its power from the vast amount of clientle on both sides of the trade. Collecting a broad set of clients, either publishers or Ad-Networks, is a complicated task and the duopoly, Google and Facebook, that currently controls this market is self evident for that.

An *Ad-Exchange* is comprised of multiple key roles, each unique and very different in nature from the others. Such roles include contacting and signing contracts with as many publishers as possible to enlarge the *Ad-Exchange*'s portfolio. Each publisher can draw different sets of audiences and *Ad-Networks* want a single *point-of-contact*, where they can reach all of their target audiences. On the other hand, an *Ad-Exchange* is attractive to *publishers* only if it can draw many *Ad-Networks* to compete for the impression opportunities they offer. And so, one of the key goals of an exchange is to establish a broad network of connections which will interact with one another. The next critical role of the *Ad-Exchange* is to run multiple concurrent live auctions, millions per second, sent by *publishers*. The entire process of getting an impression opportunity from *publishers*, offering it to *Ad-Networks* through on online auction and serving the winning ad back to the *publisher* must be fast enough for users to get the ad when a page is loading and cause a minimal delay.

The *Ad-Exchange* is, for reasons mentioned above, the core component in our model. Every agent and component directly or indirectly exchanges information with it. When users interact with publishers ad opportunities are generated and handled by the Exchange. Among its responsibilities is the facilitation of auctions between competing Ad-Networks over the aforementioned ad opportunities, the generation of daily performance reports to Ad-Networks and advertisers, to serve ads, and to abide by the limitations or requests set by publishers over how auctions should be performed.

2.1.5 Game flow: From User to Ad

An exemplar journey of a single user's visit to a website might look as follows. The user browses to its favorite news website and it's main page starts loading. At the same time the content is uploaded to the user's web browser a request is sent to the site's *Ad-Exchange* with an impression opportunity. The Ad-Exchange soon after contacts multiple Ad-Networks and auctions the opportunity to present an ad to said user. The user is not directly identified by the Ad-Exchange, but Ad-Networks can use external classification services to better identify the user. After each network determined how valuable this opportunity is to one of its campaigns, signed with advertisers, the Ad-Network bids and the Ad-Exchange performs a GSP auction to determine the winner of the auction. The winner is cross checked against the reserve price the publisher sent alongside the impression opportunity and after successful validation the ad is sent back to the user's web browser to be presented in all its glory.

In our model we tried to mimic this behavior as close as possible to the real world but some adaptations were made for simplicity. Only one Ad-Exchange is used throughout the simulation and only one classification service is offered to the Ad-Networks. Due to performance constraints, one of the limitations we put on the distributed communication model between *Ad-Networks* and the *Ad-Exchange* is that auctions will not be *live* and communicated to every participating *Ad-Network*. Ad-Networks will send a daily *answer* table with their preference on how they would bid for every possible impression opportunity presented to them. At the end of the daily simulation the AdNeworks would get the auction results and could update their bids for the next day.

2.2 Extended model

Using the simplified model allows us to generate a basic implementation for the Ad-Exchange domain we are looking to model. but, it does not fully embody the essence of a real life environment that is constantly changing, adapting to new behaviors and challenging players to compete in it. What the basic model lacks can be attributed to the incentives each player works by. Why do Ad-Networks compete against one another? Who supplies the demand for advertising? Why are publishers interested in reserve price manipulation? and many more questions we wish to answer with the extended model. This model will connect our simplified view of the domain with the TAC world, introduce new mechanisms which will drive the key players, presented in the previous section, to pro-actively participate in the exchange, and set standardized measures by which player and algorithm performance, either implemented by us or by external groups, can be compared and evaluated.

Relying on a single type of goods to be auctioned and traded restricts the simulation and the players, implementing Ad-Networks, and sets too much focus on the ad auction while ignoring other parts of the trade, such as campaign delivery and user classification, that take place either before or after the ad auction itself. Therefore we chose to introduce multiple types of goods that are auctioned during the life-cycle of the simulation. These goods are intertwined with one another in nature, one must be bought to be able to deal with an other, and users are forced to develop strategies for resource acquisition across multiple markets. Thus users are no longer bound by a single type of auctioned goods.

The next challenge is to generate a model that evaluates performance in the simulation. How do we measure if an Ad-Network is *better* than others? Having developed that model, can we make sure that networks compete throughout the simulation, and not achieve a good ranking at the start of the simulation and stop playing? How do we deter agents from acquiring more campaigns than they can handle and how do we reward or punish Ad-Networks at the end of campaigns, once they can be fully evaluated according to their performance?

These questions led us to develop a comprehensive set of mechanisms for agent evaluation, reward and punishment, for resource acquisition, and for overall examination of the simulation for future research. In the following sections we will show some of these elements, among them are the *User Classification Service*, *Campaigns* and *TAC-ADX Ad-Networks*.

2.2.1 TAC

Laying the foundation for a distributed pluggable live simulation is not a simple task, and for that we used the work done by the TAC-AA developers and TAC competitions before that. Their foundation supplied us with a stable environment to build from and allowed us to focus on the implementation of the TAC-ADX specifics.

The platform sets the basis for the development of simulations that require: time and interval tracking mechanisms, message transportation and serialization, remote and local actors which can connect to a main server and competition scheduling mechanisms which track and coordinate multiple agents across multiple simulations in order to evaluate them.

2.2.2 User Classification Service

User Classification Services, UCS, also known as cookie matching or attribution services, serve a fundamental role in the online advertising trade. Correctly identifying, segmenting or any other type of user attribution allows Ad-Networks reach their desired audience while using less resources over unknown audiences. These services, like the *Ad-Exchange* itself, can only perform well if they are deployed across a vast network of publishers. Due to their passive nature with regard to interaction with publishers, they only require the publishers to install a small piece of code in their platform to let the UCS know when a user visits. It is common to see multiple services installed on a single publisher. Integration with the Ad-Exchange service is nonetheless important since it is required to forward a user identifier for the advertising network when requesting a bid for a user impression.

Once a UCS is installed in large enough set of publishers it can follow users as they traverse from one site to another, build profiles for users according their activities on the sites and sell that information to whomever is willing to pay. Such profiles would most likely include data relevant to the UCS's customers, the Ad-Networks, and so the more detailed and segmented data a UCS can provide, its product's value for the Ad-Network increases. These services take a large role not only in providing user segmentation for ad auctions, they also identify users across platforms and enable advertisers and companies to identify who uses their products and how they got to use it, be it an ad on a popular site, a scanned QR code or any other type of promotion.

In our model we use a single UCS to provide user segmentation to Ad-Networks. This omnipotent service always knows every user's attributes and is willing to share that data to the highest bidder. The UCS auctions its service among Ad-Networks daily using a GSP auction. The ranking of Ad-Networks after the auction is completed determines which percentage of the impression opportunities will be classified, revealed fully, for them. The top bidder will have all impressions classified, the second bidder will have a lower probability denoted by the parameter $P_{UserRevelation}$, the third bidder has probability $P^2_{UserRevelation}$ and so on.

2.2.3 Advertising Campaigns

Online advertising campaigns are those who introduce money into this industry. Companies who wish to advance their brand or product will create a marketing campaign, define its goals, create visual and textual ads and will then turn to Ad-Networks to execute it in the online ad market. The advertising campaigns are the standard contract signed between an advertiser and an Ad-Network. The contract defines which ads should be shown to users, from which demographics, how often, and at what cost. Ad-Networks differ in the granularity they offer with regard to the target audience, ad serving type, pricing model ,and successful campaign execution.

Campaigns are usually defined by their target audience, budget, and expected reach, which in turn can be divided into impressions, clicks, conversions or any other style of measurement for how successful a campaign or ad was. Some of these measurements require third party service providers to link data across platforms and serve a coherent view to advertisers.

In our model we focused on campaigns that only require impressions, with no regard to clicks or conversions. The key properties a campaign is built from are the expected target audience, by size and attribution, a preferred ad type, video or text, and a time frame for which the campaign is relevant. Campaigns are allocated to Ad-Networks in the form of a GSP auction where bidders bid on the opportunity to execute the campaign in the lowest cost for the advertiser. Ad-Network bids are normalized with regard to past performance to encourage successful campaign delivery and deter bad-performing Ad-Networks from winning campaign opportunities by offering low prices.

Campaign execution constitutes of a daily effort from the Ad-Networks to bid on impression opportunities offered by the Ad-Exchange. An Ad-Network will send a daily answer sheet to the

Ad-Exchange detailing how it would bid for every impression opportunity presented to it. These answer sheets allow compound answers for campaigns targeting overlapping market segments and for unclassified impressions. The Ad-Exchange will use all the data sheets to simulate a single day and deliver performance reports back to the Ad-Networks.

2.2.4 Demand Agent

The cycle of campaign generation, execution, reporting and tracking is often carried out my multiple entities trading with each other. Advertisers generate the demand for ads to be displayed, Ad-Networks carry out the daily work of publishing these ads, Ad-Exchanges generate performance reports, and aggregation companies summarize these reports for the Ad-Networks and advertisers.

Under the TAC-ADX model we implemented a unified agent that acts on behalf of these multiple, sometimes adversarial, entities. The demand agent acts as a multiple service platform, starting with generating demand. Campaigns, offered by advertisers, are created on a daily, simulation-wise, basis and auctioned to competing Ad-Networks. The demand agent uses data from the performance tracking module to estimate how likely an Ad-Network is to execute a campaign and this data is taken into account during the campaign auction. Though this information can be exchanged between the tracker and the campaign generator we are implementing a cooperating entities platform currently. A future extension for the model might consider trading this information as a type of goods.

Reporting and tracking is another service provided by the demand agent to Ad-Networks. The constant flow of events and messages passing in the system is constantly monitored by the demand agent. Due to performance issues it is not feasible to deliver each of them to remote Ad-Networks, therefore the demand agent will aggregate and extract important data for each Ad-Network individually, at no cost to them, and supply daily reports for their use.

Chapter 3

Reserve Price

3.1 Auctions

Auctions have a long history and are a common practice for both product valuation and trade. It introduces, depending on practices used, a standard set of rules by which product pricing is achieved, Buyers are allocated goods and trade rules set by either party are enforced. The practice of auctions includes a *seller* offering a product for sale, *buyers* who wish to acquire the product and an *auctioneer* orchestrating the process. The auctioneer retrieves bids from the buyers and. according to the auction type, at the end of the bidding process the product is either sold or not. Bids can either be *sealed bids*, where buyers are unaware of their opponents bids, or *open bids*, where the buyers observe other buyers' bids. In the process of the auction, the bids can be ranked in a descending or ascending order, a buyer might be limited to a single bid or be allowed to submit bids multiple times. Additionally we might have more modifiers applied to the bids for the auction resolution process, regarding prices and allocation.

The best known version is an *English auction* where the auctioneer starts at some price, and the buyers submit open bids for a product. At any time, the tentative winner of the auction is the buyer with the highest bid, and other buyers can outbid him, by offering a higher price. When no more bids are sent to the auctioneer the auction concludes and the product is sold to the highest bidder for a price which is their bid. A variation of this auction type can include a reserve price which is a minimum price the seller is willing to accept for the auctioned good, and in that case, the auctioneer will start the bidding at the reserve price. If there is no bid above the reserve price, the product remains unsold. Another auction type is the *Ducth auction* by which the auctioneer starts with an initial high price for the auctioned product. The auctioneer decreases the price gradually until some buyer is willing to pay that amount. At such a time the auction terminates and that buyer receive the product at that price. If there are multiple identical products, the process continues until all of the goods are sold or until the current price equals to the reserve price, under which the seller is not willing to go. The equivalent sealed bid auctions are: sealed bid first price auction (equivalent to Dutch auction) and second price sealed bid auction (which is similar, but not equivalent to an English auction).

Sealed bid second price auctions, also called Vickrey auctions, are another well known type where bids are submitted once to the auctioneer, and not observed by the other buyers. The winner is the buyer with highest bid and the paid price will be that of the second highest bid. Vickrey auction incentivizes buyers to bid their true valuation for the product at sale, assuming no future repeated interaction between the buyers, and independent private valuation of each buyer to the product auctioned.

A widely used extension of the Vikrey auction in the Ad-Exchange domain is the *Generalized second price (GSP)*. In a GSP auction more than one product, a.k.a slot, can be put up for sale, but each buyer can be interested in at most one product (slot). Buyers place their bids and these are ranked in a descending order. The first slot is allocated to the highest bidder, the second slot is allocated to the second highest bidder, up to the number of slots or bidders. Each buyer pays the minimal amount that is required to secure that slot, given the other buyers bids. This implies that the highest bidder will pay for its slot the second highest bid, the second highest bidder will pay for its slot the third highest bid and so on. Unlike the case of a single product in the Vickery auction, in GSP there is no dominant strategy for the buyers, and buyers might gain by not bidding their true valuations. There is a vast literature on the topic of GSP and its incentives (see, [2, 14]).

3.2 Reserve Price

Any of the auctions type referenced above can be modified by the seller to increase its expected revenue from auctioned products. A common practice is to introduce a *reserve price*. This concept is possible only if the seller is willing to accept an outcome by which the product remains unsold. This is a reasonable assumption if the product is valued by the seller at a certain price larger than 0. Another case by which the seller might use a reserve price is to influence the buyers to bid higher. An extreme case is when there is a single buyer and the reserve price becomes the price of the item. Another incentive for the seller is to influence future auctions and increase the expected revenue from them.

A reserve price in an English or a Vickrey auction implies that the product is sold to the buyer only if the highest bid is higher than the reserve price. In a Vickrey auction the reserve price can have two effects. First, if it is higher than the highest bid, the product remains unsold. Second, if the reserve price is between the highest bid and the second highest bid, then the winning buyer will pay the reserve price. In all other cases, the reserve price does not influence the auction.

The introduction of a reserve price can have different effects on different auctions types. In a Vickrey auction, where a dominant buyer's strategy is truthful bidding, the reserve price can be considered as an additional virtual buyer, and therefore it remains a dominant strategy to bid one's valuation.

In the Ad-Exchange the auctions are GSP, where truthful bidding is not a dominant strategy in any case, and sellers can allow unsold slots in auctions. This implies that we enter an interesting realm of a "mind-play" between sellers and buyers. The product being sold is ad space and the revenue the sellers wish to increase is the expected sum of the revenue from all their sales, and not the expected value of a single product sale. Thus, reserve price manipulation strategies take an important role in this market and their effect can result in a great increase to the sellers revenue.

3.3 Reserve Prices for TAC-ADX

Upon building the TAC-ADX infrastructure and running multiple competitions with a variety of agents, each having a unique approach for bidding and competing in the simulation we looked for a way to harvest the platform for research of internal mechanisms. We sought to use the different implementations of Ad-Networks that were developed in the past and so we focused on components that these Ad-Networks interact with. We needed a module that has a utility function we can try to increase.

The selected module was the reserve price selection mechanism. This module is in charge of generating reserve prices for every impression opportunity sent from the publishers. Its utility function is the sum of all revenues generated by the second price auctions throughout a simulation. Therefore we examined algorithms generating reserve prices for the second price auctions aiming to increase the generated revenue.

The data we wanted to feed into the algorithms was based on historical bids and auctions performed in a single simulation. Since different simulations can have different publishers, initial values, and participating agents we felt there is little sense in feeding data from different simulations to whichever algorithm we will use.

The results show that using the DC algorithm proved to be very capable in doubling the revenue limits reached by both the static and dynamic approaches.

3.3.1 Historical Data

Available data for training the different models can range from every auction ever run under the TAC-ADX platform to only a few of the last auction performed during a simulation's day. Though, different simulations can have different randomized attributes generated for them and so we need to address them if we intend to aggregate data from multiple simulations or perhaps ignore them if we train our models in-game and do not carry data from one simulation to the other. Another aspect that can affect learning guarantees is that each simulation can have different participating Ad-Networks.

Three approaches will be presented next and they all rely on limited data presented to them, available only during a simulation. They are not allowed to transfer "knowledge" between simulations. For every impression opportunity auction available data includes all bids and all qualifying attributes for the auction's query.

3.3.2 Static values

The first algorithm we will test is the easiest to implement - setting static values for the *reserve price*. We are limited to a single degree of freedom using this method and that is the initial value. A *zero reserve price* is a special case where we let the market decide on the product price without intervening. This method can be useful to compare others against due to its simplicity.

We will use multiple static reserve prices starting from 0, growing linearly. Assuming that the generated revenue will increase in correlation to the reserve price we expect to see an increase in revenues and then a drop, since Ad-Networks won't be able to bid these high prices for individual ad spots without damaging their overall performance. This approach doesn't try to maximize revenues by inspecting individual auctions and determining the highest reserve price possible for the auction, where $r <= b_1$, it focuses on raising the global price for ad spots by depriving Ad-Networks from winning any ad spots for using low prices.

3.3.3 Dynamic values

Every TAC-ADX simulation is initialized and run under changing conditions such as: campaign lengths, targets, allocations, etc. We want to explore if given a current state of the simulation we can learn a local-maximum to increase revenues by inspecting the last n transactions for a given product. For example, all auctions for *High Salary Females browsing Yahoo* performed in the last day. We want to explore which reserve price produces a higher revenue for this cross-section and use it in the next measurement period. This iterative learning process can update the reserve price baseline towards high revenue reserve prices. Though we do not inspect a global maximum we expect the changing baseline to adapt itself to the market and therefore generate higher revenues than those generated by the static approach.

For any given set of features, *Publisher* $\times$ *User type* $\times$ *Impression type*, the algorithm starts with a reserve price that is randomly chosen from a uniform distribution between two initial limits which we will mark as $b_1(u, a)$. For a given day t, the algorithm will produce different reserve prices $r_t^i(u, a)$ by selecting values around the daily *BaseLine* $b_t(u, a)$ with a normally distributed perturbation with a zero mean and a defined variance. At the end of the day the algorithm will shift the BaseLine towards the reserve price that averaged the highest profits $r_t^{max}(u, a)$ for that day with a given learn rate γ.

$r_t^i(u, a) = b_t(u, a) + \epsilon_i$

$b_{t+1}(u, a) = \gamma \cdot b_t(u, a) + (1 - \gamma) \cdot b_t^{max}(u, a)$

Variations for the algorithm's implementation include changing how often a new baseline is derived, which starting position we use, how wide is the range we select new values from, and how fast we wish to update the baseline towards new values while remembering old ones.

3.3.4 Difference of Convex

Previous research on reserve price optimization was done by [10] over an existing data set for sponsored search ads where queries were grouped according to keywords and an optimal reserve price was calculated per group. Another approach given by [3] assumes limited visibility of the data by the seller and uses a regret minimization algorithm for reserve price selection. Upper bound and heuristic lower bound on expected revenue in auctions with convex perceived payments were researched by [5].

With the help of the TAC-ADX platform we want to explore if we can optimize the reserve price under a complete visibility of past auctions. That includes features for each auction and a complete bid list. For that we turned to the DC, difference of convex, functions programming approach which relies on a method described in [13] and [7] for revenue maximization in GSP auctions with a reserve price. In [7] the authors describe at first an algorithm for the no-feature case and continue to extend it to the general case where features are present. For a given set of auctions $A_1, ...A_n$ with features $V_1, ...V_m$ and bids $(b^{11}, b^{12}), ..., (b^{n1}, b^{n2})$ the algorithm computes coefficients $C = (c_1, ..., c_m)$ in order to maximize $\sum_i \text{Rev}\left(r_i, b^{i1}, b^{i2}\right)$ where the reserve price is $r_i = V^i \cdot C$, and b^{i1} and b^{i2} are the highest and second highest bids at auction A_i. (Recall that $\sum_i \text{Rev}\left(r, b^1, b^2\right)$ is: (1) r if $r \in [b^1, b^2$, (2) b^2 if $b^2 > r$ and (3) 0 if $r > b^1$.

For each auction, the two highest bid prices as given by the Ad-Networks and features from the auction's query were normalized and fed to the model. To allow proper training the first 10 days of every simulation relied on the TAC-ADX original adjustable moving baseline, as described in [11], and from day 11 to 60 we used the results of the DC algorithm to generate reserve prices.

Some of the challenges we faced while implementing the algorithm as a TAC-ADX module involved running it under a constrained environment where its results needed to be calculated within a limited time frame for the game to run as scheduled without requiring the agents to wait an un-natural period, under the simulation rule set, for the results.

Determining the training data set size was another challenge we needed to understand. Because a simulation lasts 60 days, and we cannot train on datasets from previous simulations we needed to slice the data set into two parts: training and test. By exploring the relation between training length to performance we identified that a learning period of $8 - 12$ days at the beginning of the simulation is sufficient for achieving high revenue. After that, any extra days, including up to 40 learning days, did not yield an improvement of more than 5%. Therefore we selected the 10 days as the separator between the learning period and using the learned coefficients.

3.3.5 Results

Analyzing millions of impression opportunities, bids, features and reserve prices requires a standardized comparison, accumulation, and reduction method of the data for visualization and ease of understanding. For that purpose, each simulation exports a single file containing every auction that has taken place during its 60 days period. After all simulations ran successfully a parsing tool, written efficiently and solely for the purpose, was used to calculate a daily accumulated revenue for all auctions performed in every simulation. The revenue for a single auction is based on the formula for a GSP auction.

Static values

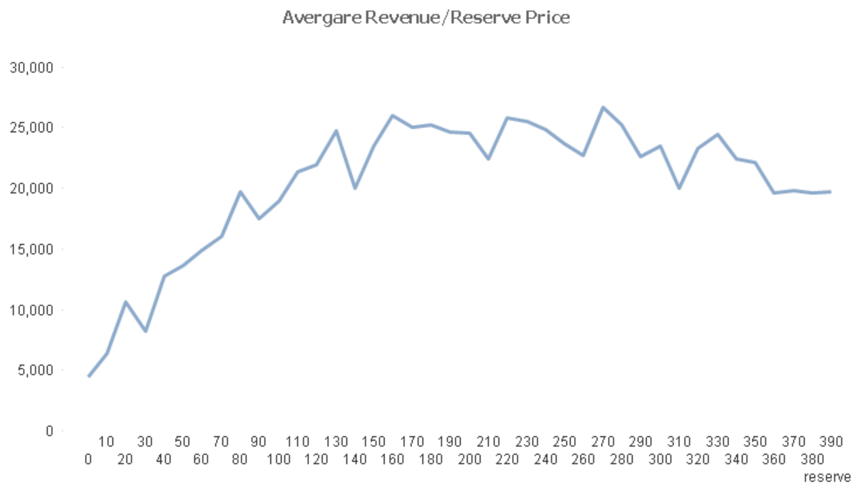

Figure 3.1: Comparing average total simulation revenue (y-axis) for varying values of static reserve prices (x-axis)

We start by examining the difference between the average global revenue for publishers over a course of multiple simulations played with varying reserve prices as shown in Figure 3.1 [1] [2]. Every simulation was classified according to its active static reserve price and every group of simulations sharing a single reserve price was averaged to calculate the group's average revenue per simulation.

It is visible that, mostly, in the range of 0-160 the revenue function is an increasing monotonic where revenue increases the higher reserve price is in effect. Beyond that range, from reserve price 160 and onwards, there was a constant decrease in global revenue but at a lower gradient. Examining two ranges with specific significant reserve prices produces a clearer image. In the range of 0-160 higher reserve price not only yielded global higher revenues as shown in Figure 3.2, but also over-took lower reserve prices revenues on a daily basis for almost every day in the simulation, Figure 3.3. This simulation behavior is as expected from deploying an incremental reserve price change over the course of multiple simulations. Higher reserve prices can increase publishers' revenue by utilizing several auction types. Auctions having a single bidder can generate higher revenues as long as the static reserve price is not higher than the buyer's valuation price since the resolved price will be the reserve price itself. Auctions already having a competition from several parties can now be more competitive by introducing a new shadow buyer, the reserve price, which can out-bid other buyers and increase the products value artificially for the other members.

[1] For ease of understanding reserve prices values are shown with a 100K multiplier

[2] All revenues were calculated starting from simulations' 10th day to remove any first days' chaotic behavior related affects

Figure 3.2: Comparing average total simulation revenue (y-axis) for varying values of static reserve prices (x-axis)

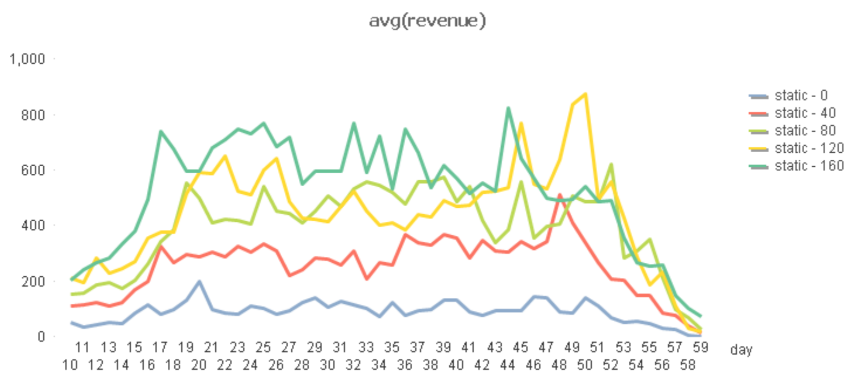

Figure 3.3: Comparing average daily simulation revenue (y-axis) for every day of the simulation (x-axis) for several static reserve prices (colored lines).

When we continue to examine performance for selected reserve prices in the range 160-360 the change in generated revenue is still visible, but in a much milder manner. A decrease in the global revenue can be seen in Figure 3.4 as reserve prices increase. Unlike Figure 3.3 we cannot directly differentiate each reserve price performance when comparing them on a daily basis in Figure 3.5. This decrease in revenue can be explained through the eyes of the agents - with each higher static reserve price product prices continue to increase while the agent revenues do not. And so, bidding higher prices is not as valuable as before and agents are less competitive.

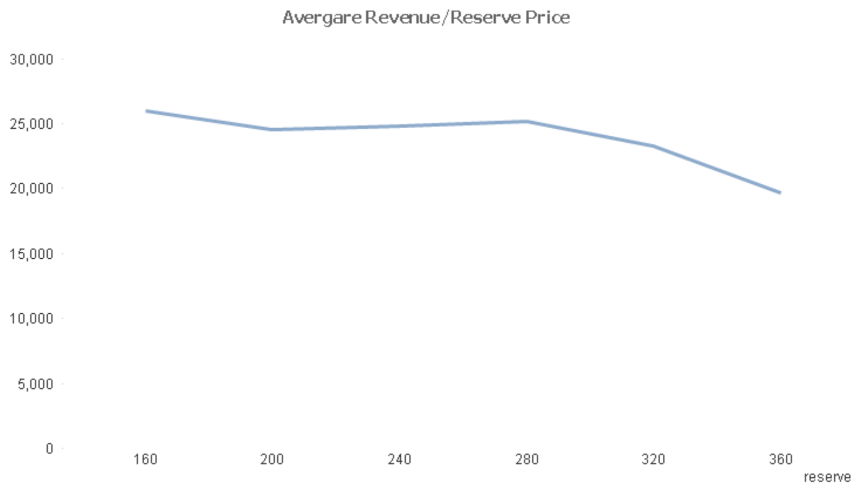

Figure 3.4: Comparing average total simulation revenue (y-axis) for varying values of static reserve prices (x-axis)

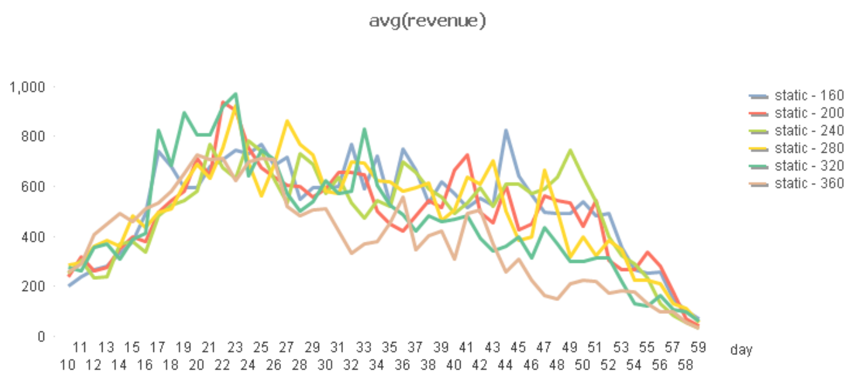

Figure 3.5: Comparing average daily simulation revenue (y-axis) for every day of the simulation (x-axis) for several static reserve prices (colored lines).

Dynamic values

Exploring different options for dynamic value parameters involves a 3-dimensional matrix which we can play with. The first parameter is the initial reserve price baseline which can be used to push the algorithm towards higher reserves if we think it is not climbing high enough in its learning process. The second parameter controls the explore range around the daily baseline the algorithm will generate different reserve in and asses which reserve price yielded the highest revenue. The last parameter is the learn rate which controls how fast we shift from our current daily baseline towards the daily best reserve.

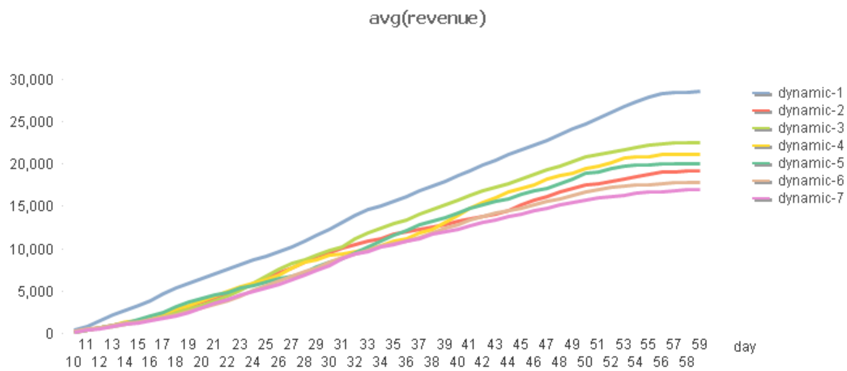

Figure 3.6: Comparing average daily simulation revenue (y-axis) for every day of the simulation (x-axis) for several dynamic initialization parameters (colored lines).

As can be seen in Figure 3.6 the first approach which used the original parameters published in [11] suppressed all other approaches. Since agents used for the simulation trained against this approach solely they might have fitted themselves to its generated reserves and therefore the simulation product prices fitted their model best when this approach was active.

Our next target is to compare and understand the difference between the performance of the best performing dynamic approach to the best static. Figure 3.7 shows that both approaches performed very similar to each other with a slight advantage to the dynamic approach. A different view of the data, as shown in Figure 3.8 with no daily accumulation, shows where this difference is derived from. During the first stage of the simulation where most of the agents are still active and the last stage where only the best agents, in a given simulation, are left the dynamic approach out-performs the static's performance. The first stage can be explained by having the dynamic approach better fit prices given in a competitive market and having the ability to adjust to market forces influencing demand and supply. In the final stage, where competition is dull the dynamic approach can lift product prices by introducing an artificial demand that adjusts to the agents and increase the more they react to it, unlike the static reserve price which does not utilize these days for a higher revenue.

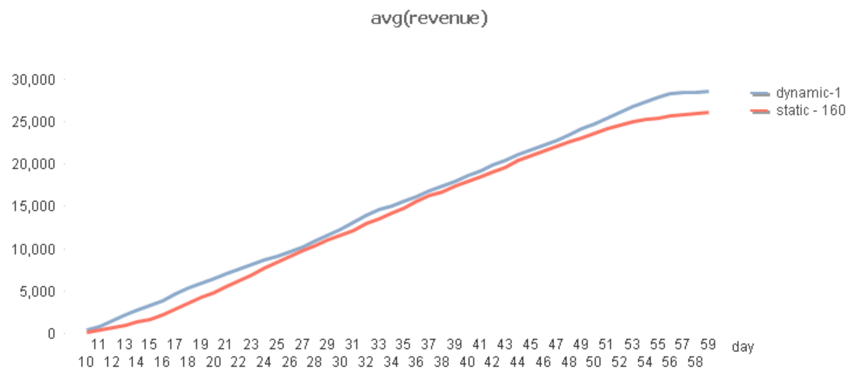

Figure 3.7: Comparing average daily simulation revenue (y-axis) for every day of the simulation (x-axis) for best dynamic and static approaches (colored lines).

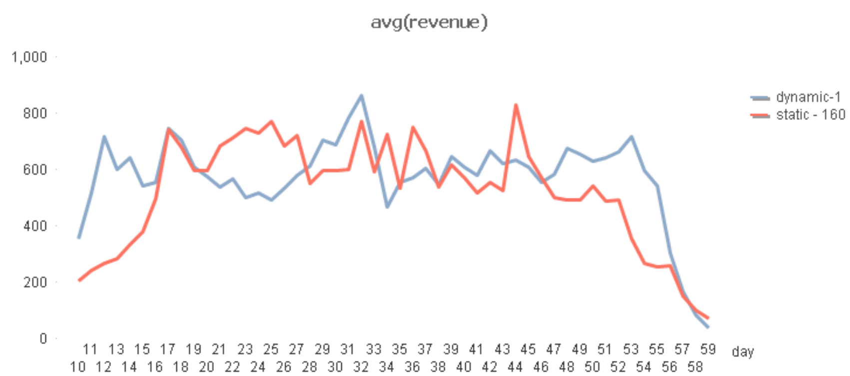

Figure 3.8: Comparing average daily simulation revenue (y-axis) for every day of the simulation (x-axis) for best dynamic and static approaches (colored lines).

Difference of Convex

The last stage of the comparison introduces the DC algorithm where the dynamic approach is in use in every simulation up until the 10th day and from then on up until the last day the calculated coefficients are applied and the reserve price is derived from them for every auction. For every publisher a different DC context was kept, with bids and features from past auctions including Gender, Age, Income, Device and Ad Type. Figure 3.9 shows that the DC algorithm is far more effective than both the dynamic and static algorithms. A non-accumulated view, as shown in Figure 3.10, shows that the DC approach out-performed all other approaches on a daily basis, with no regard to the different stages of the simulation.

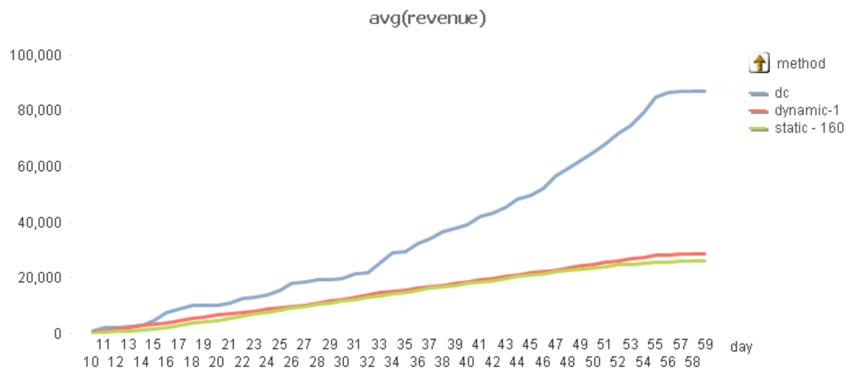

Figure 3.9: Comparing average daily simulation revenue (y-axis) for every day of the simulation (x-axis) for best dynamic, static and DC approaches (colored lines).

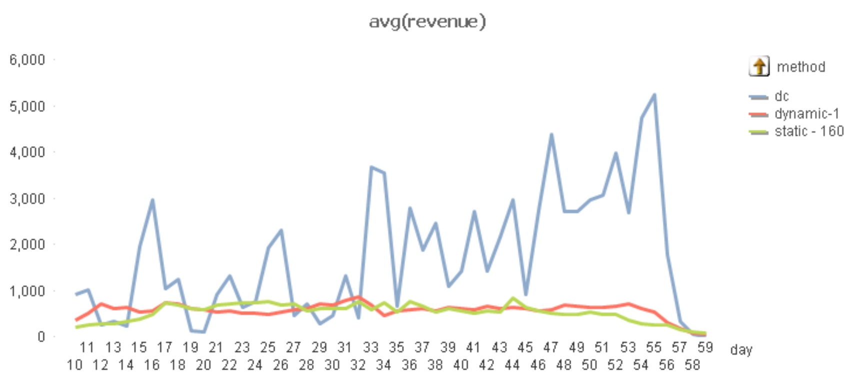

Figure 3.10: Comparing average daily simulation revenue (y-axis) for every day of the simulation (x-axis) for best dynamic, static and DC approaches (colored lines).

3.3.6 Conclusions

The TAC-ADX model and implementation proved to be a capable tool for research, exploration and understanding of the online Ad-Exchange environment. Starting from a simple competitive agent implementation, to playing against the many different agents who interact and influence the game and even changing game mechanics and witnessing its effects the platform enables the audience to engage in this vast domain.

This platform is a work of not only the writers of this paper, but of the students, researchers and partners who helped us develop, test and extend it, have devised intricate algorithms and agents to compete in it and helped advance our model from a TAC simulation to a great echo system we can explore.

Using our models we have shown that reserve price manipulation can influence and improve publishers' revenue when applied to the simulation built under the ADX model. Though simulations can differ wildly from one another a standardized normalization can allow researchers to compare different algorithm influencing their outcome. The DC approach which introduces a quadratic programming equivalent to finding coefficients for auction related features to determine a reserve price for auctions prevailed all other methods and more than doubled their revenue. A study group size of 10 days was more than enough to run the DC algorithm and use its result for the remaining 50 days.

Future research can explore different mixtures of the presented approaches which can be applied throughout different simulation stages. As we can see in Figure 3.10 the DC approach performs very closely to both the static and dynamic approaches in the range of days between 10-30 and only exceeds their average revenues from day 30 and onward. Perhaps training on different days, other than 1-10, extending or decreasing the training day count or even using an alternating approach between DC and others can yield better results.

Another interesting issue that comes to mind is the fact that we could not transfer data learned from one simulation to another when using the DC approach. This can be attributed to missing features that are currently overlooked and influence the outcome of auctions, e.g, random simulation parameters.

Chapter 4

ADX Game implementation

4.0.1 Infrastructure

The AdX simulation uses a modular architecture which allows teams from around the world to implement their own designs for profitable *Ad-Networks* and test them against other players. Relying on the architecture developed for TAC-AA we utilized existing protocols, algorithms, and mechanics allowed us to focus our resources to essential ADX related parts without diverting our attention to the development of these tools.

The system is built in a *client-server* manner where a single main server runs the core components for the simulation and clients connect to it remotely for interaction. Though the server is intended to run on a remote node accessible to all players it can be easily deployed by participating teams locally for ease of development. Core features of the server such as the *Demand Agent* and the *AdX* behave as local agents and share the same messaging platform used by remote agents for communication. The main server serves multiple purposes, as shown in Figure 4.1: coordinating and supporting communication of local and remote agents, simulation and competition tracking and scheduling, Figure 4.2, user management and an online view where participants can track performance of ongoing simulations.

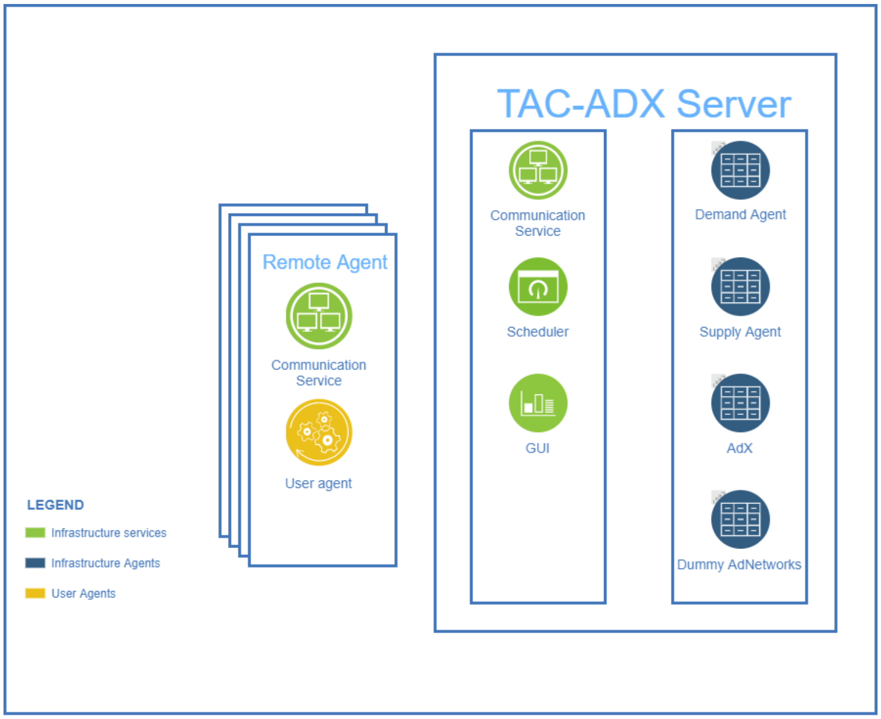

Figure 4.1: TAC-ADX Components.

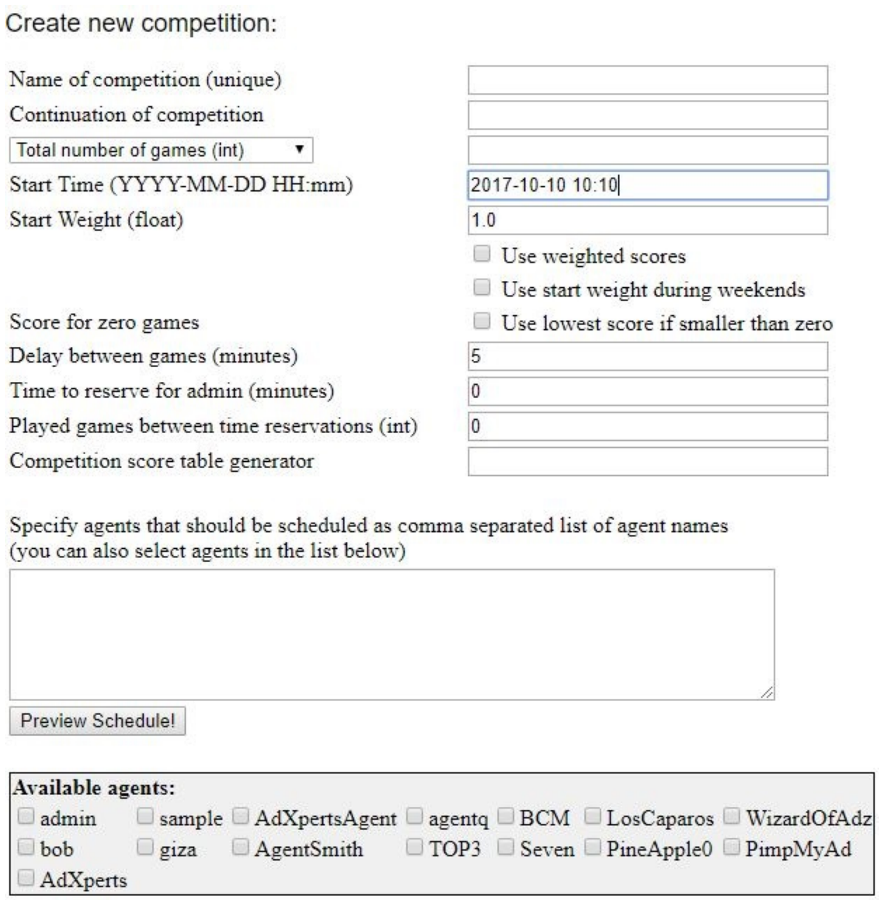

Figure 4.2: Competition scheduling form.

Communication Service

Module responsible for message serialization and deserialization, agent connectivity to the main server, including remote and local agents, and network flattening. This module is installed in the TAC-ADX server and in any agent that wishes to connect to it. Once a remote agent is connected to the main server, a publish-subscribe interface is presented to it, and the agent can receive and send messages over this channel as if the other parties were directly connected to it. Most messages are routed in one of three forms: (1) direct agent messages are sent from one agent to another (2) group broadcast messages are sent to agents based on their classification (3) multiple classification messages are applicable per agent, e.g, group of Ad-Network agents, group of remote agents..

Scheduler

There are two main scheduling environments used in TAC-ADX: Global and per simulation. *Global scheduling* enables remote agents connected to the server to set up simulations and coordinate their timing across all participants. Under the standard protocol as long as any agent is connected to the main server a competition is either taking place or scheduled to start in the coming minutes. Agents who missed registration for a running simulation can wait and they will be automatically joined to the next one. A competition is a group of 40 concurrent games scheduled under the main server with pre-defined remote agents allowed to participate, as can be seen in Figure 4.3.

Simulation Scheduling controls the timing and flow of messages in a specific simulation. A simulation is built from 60 *days* where every day has a length of 10 seconds. The scheduler tracks time and every 10 seconds sends a *tick* messages to all agents in order to keep them synchronized. The agents soon-after update internal state to reflect that a new day has started and send messages accordingly. Local server agents, such as the demand and supply, are given unique *sub-ticks*. These enable them to have a higher granularity of time-keeping. They also allow them to run house-keeping tasks either just before a new day starts, or as soon as it ends, e.g, building daily reports. As a consequence remote agents will only interact with the main server after they received specific request messages sent from local agents, such as a campaign opportunity message, and will not directly act when they get the new day *tick*.

ID	Time	Type	Participants	Status	Join
			Competition Demo begins (2018-01-26 10:10:00)		
	10:10:00- 10:20:00	tac13adx	agentq, BCM, LosCaparos, WizardOfAdz, giza, AgentSmith, TOP3, AdXperts	Coming	
	10:25:00- 10:35:00	tac13adx	agentq, BCM, LosCaparos, WizardOfAdz, giza, AgentSmith, TOP3, AdXperts	Coming	
	10:40:00- 10:50:00	tac13adx	agentq, BCM, LosCaparos, WizardOfAdz, giza, AgentSmith, TOP3, AdXperts	Coming	
	10:55:00- 11:05:00	tac13adx	agentq, BCM, LosCaparos, WizardOfAdz, giza, AgentSmith, TOP3, AdXperts	Coming	
	11:10:00- 11:20:00	tac13adx	agentq, BCM, LosCaparos, WizardOfAdz, giza, AgentSmith, TOP3, AdXperts	Coming	
	11:25:00- 11:35:00	tac13adx	agentq, BCM, LosCaparos, WizardOfAdz, giza, AgentSmith, TOP3, AdXperts	Coming	

Figure 4.3: Scheduled competition with time and participants.

Graphical User Interface

The Graphical User Interface, *GUI*, is intended for data visualization of the running simulation. Contestants can use it track performance of their agents, see Figure 4.4, and also track other agents. Additionally it allows them to follow ongoing events and explore agent stats, see Figure 4.5. This interface is only intended for human users since some of the exposed data, that users can see, is hidden from the actual electronic agents and should not be incorporated into their logic when calculating their next move. This compartmentalization is enforced in the remote agent software but players are assumed to play fairly.

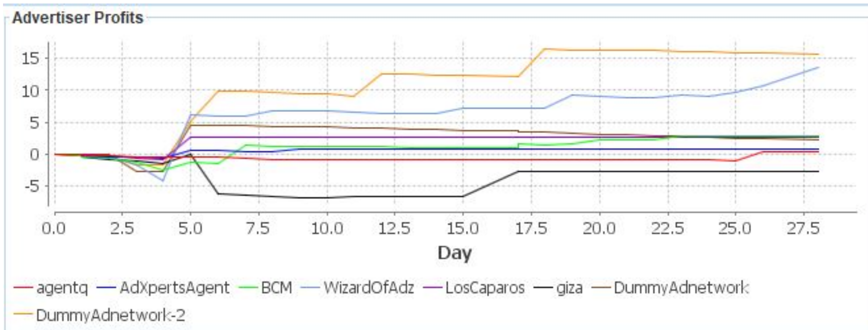

Figure 4.4: Agent revenue tab.

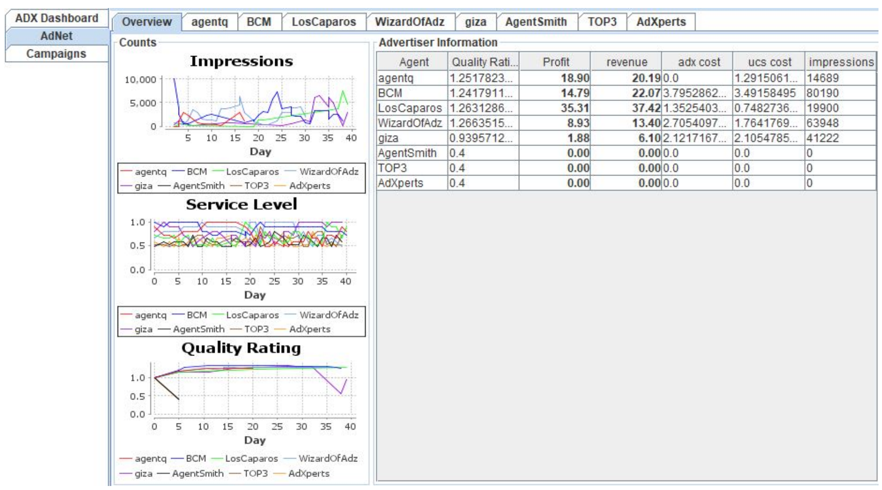

Figure 4.5: Agent overview tab.

Demand Agent

Generates campaign auctions, tracks agent performance throughout the game and manages the *UCS*. A daily campaign is generated by the agent and auctioned among the Ad-Networks. Once a campaign is allocated the Demand Agent will track the Ad-Network performance and use it to calculate the Ad-Network's quality rating. This rating will influence how future campaigns are allocated and auctioned to Ad-Networks but will not be applied to other auctions such as ad-auctions or UCS. The UCS is a complimentary service managed by the Demand Agent responsible for supplying Ad-Networks with a paid classification service for ad-auctions. Every day the Demand Agent will auction its classification services with no regard to past performance. This is in contrast to the campaign auctions.

Supply Agent

Generates impression opportunities by simulating user traffic to publishers. The Supply Agent will generate a daily user traffic according to the simulation's generated population. Every user from the population is characterized by features that are likely to dictate its moves. These include Age, Gender, and Income level. Publishers are divided into three non-overlapping sets. Each set contains 6 publishers. At the beginning of each simulation one of the sets is selected and the simulation is initialized from it. User browsing to these publishers is simulated throughout the day where users are more likely to browse publishers that attract them based on their defining features.

Though started as a mostly passive agent with limited responsibilities, e.g, handling daily user traffic and contacting the Ad-Exchange with impression opportunities, the agent evolved during the game's lifetime. The third game's yearly revision focused on publishers' revenue and this domain eventually became a main research point for this thesis.

Ad-Exchange

Brokers auctions between Ad-Networks for impression opportunities. Decisions made in the process of designing both the simplified model and the extended model led to the design of a unique Ad-Exchange with added capabilities and responsibilities. Unlike actual Ad-Exchanges our implementation is not able to contact interested Ad-Networks for every impression opportunity that arrives, and so Ad-Networks send a daily response sheet for every question the Ad-Exchange can ask them that day. This led to the extension of the Ad-Exchange which incorporated a simulation of the Ad-Networks in the game. This simulation tracks the progression of every auction throughout a day and bids according to the instructions sent by the Ad-Networks. These uniquely crafted messages allow Ad-Networks to define with great precision and also non-deterministically how they would like their counterpart in the simulation to act on their behalf.

Ad-Networks

Participants in the TAC-ADX competition are responsible for the implementation of the Ad-Networks. Each Ad-Network implements a basic interface to handle communication with the game server. The interaction with the game server can be grouped into two types of messages: (1) requests made by the server to the client (2) reports sent by the server to the client . While reports are uni-directional and do not require a response from the client, requests are sent due to an event triggered in the simulation and a response is expected from the client. Ad-Networks are shielded from most of the occurrences in the game and so their only gateway to the simulation is through these messages. These were built to expose just enough data for the Ad-Networks to build their own model of the current state of the simulation.

To make sure agents were written to the same standard of communication with the server, participants were asked to implement only a callback interface for every message that can be sent from the server. Doing so limits the user's ability to incorrectly interact with the infrastructure. Basic messages included a *Campaign Opportunity Message* used to inform agents of an upcoming campaign that is to be auctioned, a daily *Campaign report* detailed with statistics about how well the agent progressed in achieving campaign goals and a *Simulation Status* which triggered users to send their daily *Bid-Bundle* detailing how they would like their Ad-Exchange counterpart to be on their behalf.

Game Flow

The game starts with an initial *Campaign Allocation message* sent on day 0 for every *Ad-Network*. Each agent is granted a campaign with expected impressions, reach count, and targeted market segments. The campaign is due to start on the first day. Every day n, starting from the second, a *Campaign Report* is sent to each Ad-Network detailing its accumulated statistics, targeted and non-targeted impressions, and cost, on their active campaigns up to that day, not including. The next message, which is also sent on the first day, is the *Campaign Opportunity Message*. This message is sent by the Demand Agent to inform participants of an upcoming campaign starting at day $n+2$. Campaigns are defined by their length, target impression reach count, market segments, and their preference toward video or mobile impressions. Ad-Networks will reply to this message with their requested budget to execute the campaign and with another bid for the UCS auction. The results of both auctions are reported on the following day, $n+1$, under the *Daily Notification Message* before the next *Campaign Opportunity Message*. The last set of messages includes a balance report, global publisher report with statistics about visiting users, and and Ad-Network report with detailed performance for each sub segment of the user population and its campaigns. All messages are relevant up to the current day but not including it. Once the exchange of all these messages completed the daily simulation of user behavior is run. Along side it agents are expected to generate *Bid-Bundles* for the next day, $n+1$. A Bid-Bundle maps target market segments to bid probabilities, amounts, and prices to be used for each expected impression type. Once a *Simulation Status* is sent from the server to notify the agents that a simulation day has passed, they respond with the generated *Bid-Bundle* and the day ends.

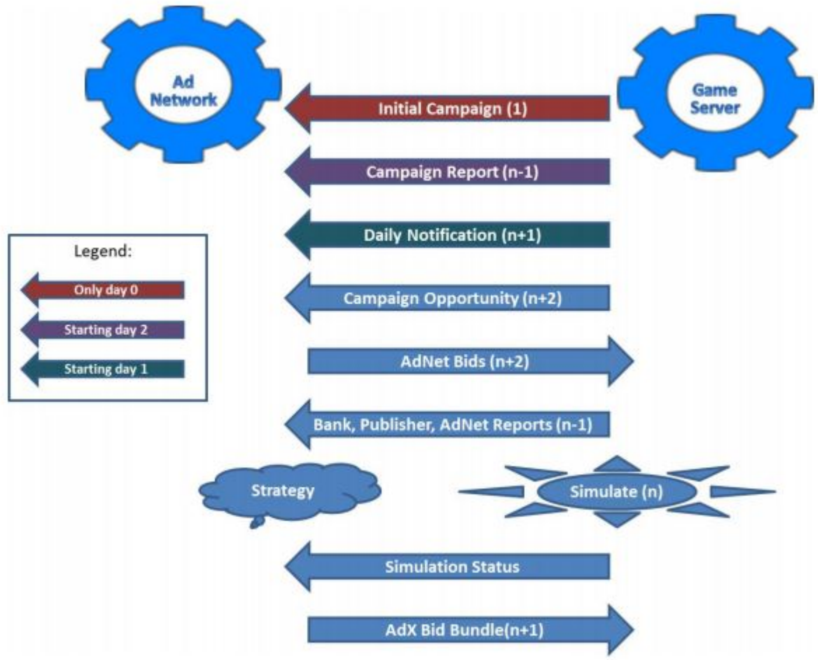

Figure 4.6: TAC-ADX daily message flow. Messages are relevant to the day number marked by parenthesis.

4.0.2 Agent strategies

Throughout the years the TAC-ADX project was developed by many teams that competed in the tournament, either internationally or at a Tel-Aviv University workshop. Multiple approaches were developed for handling certain aspects of the game.

The *User Classification Service* offers Ad-Networks the option to pay for impression opportunity classification. Though it might seem important to keep a high classification level, competing teams have developed several approaches for this service, depending on their needs and state of the simulation. During the first days of the game there is a very high demand for ad spots since every agent receives an initial contract. This demand leads to a bid-storm due to the fact that agents are not yet ranked when the game starts and their ranking for the next days will be based mostly on their initial campaign performance. Therefore, reaching a high classification level during these days is a very common strategy.

On the other end of the spectrum, agents with no active contract can choose to have no classification at all since they have no use for it. Other factors that might influence agents to target a lower classification level can include their active contracts fulfillment levels and target segments. The lower their risk is by not using the classification service can lead to lower classification level targeting.

Strategies for successful campaign allocation include: (1) bid regularization, where agents normalize their bids to adjust for their and other players' ratings (2) maintaining "desperation" levels, comprising of how much the agent needs a contract to raise its quality rating [12] (3) sending low bids in hopes for a random campaign allocation (4) looking for campaigns with a small target reach requirement that are easy to complete and improve the agent's rating.

4.0.3 Main challenges

From Paper to Code

The specification for the game was written ahead of the code that followed and implemented it. Since the game was developed for the TAC community and had a TAC-style life-cycle, relying on the existing TAC infrastructure was our first choice. Development branched from the TAC-AA which was one of the two leading branches of the TAC platform. Our first task was clearing the existing TAC-AA related code from the platform and reach a stable state where remote agents can connect to the main server and achieve an empty game life-cycle of 60 days. This simple looking task proved to be a complex one since a lot of code sections in the existing game were coupled between the TAC-AA implementation and the TAC infrastructure. Therefore we de-decoupled the code base and continued to create an empty simulation that will be the base for our TAC-ADX game.

After a basic simulation was in place, the next stage was developing the four major components: Demand Agent, Supply Agent, ADX, and a sample Ad-Network which will be able to compete in the game. This stage started with developing an MVP, Minimum Viable Product, that allows the sample Ad-Network to register a campaign with the Demand Agent, send bids to the ADX and receive simulation reports from the ADX after daily ad-auctions were simulated. Most of the component development process was independent, since we agreed on their shared interfaces before development and each component could mock others according to their interface. A key interface regarded how Ad-Networks described their intentions to the ADX every day to bid on ad auctions on their behalf. Since live bidding was not supported on the platform we wanted the agents to have a very fine grained control over the bidding strategy sent to the ADX. In the final *Bid-Bundle* structure Ad-Networks could describe for each query type, either specific, e.g, Old-Female-High Income browsing Yahoo! from mobile device watching a text ad, or broad, e.g, every Male, how much they would like to bid, which campaign should be associated with this bid, and a weight which will be taken into consideration if multiple entries are applicable for a given ad-auction.

Once all of these components interacted with each other and a simple simulation could be run where agents bid over campaigns and compete with each other, the rest of the demands from the specification were added to the code base. Among them were the *User Classification Service*, advanced reporting system to update players on their performance in the game, GUI support for live visualization, and a comprehensive testing suite to enforce the specification.

Validation

Code validation and assuring that the specification written by [11] was translated to code and protocols in a truthful manner were of high importance to us. Every promise, assumption and definition mentioned in the said paper were tested and validated using external tools. Simulation logs were recorded for every game and the built validation tool ensured specific behaviors and continuous procedures stretching over multiple simulation days kept to the standards participating Ad-Networks expected.

Figure 4.7: Example test report validating state of a specific Ad-Network on a given day.

Later on, this validation tool was given to agent developers to help them diagnose game performance and actions. Additional features focused on agent development, such as campaign validation, auction data filtering, and extraction, were added for their use. Some groups improved the tool and added custom simulation message parsing to understand internal processes. Some even incorporated the tool inside their agents to use data from previous simulations for live games, mostly in cases of competitions where 40 consecutive games take place.

Preparing for the Unknown

While developing the platform we tried to anticipate different behaviors, actions and interactions remote agents will have with the ADX server. For that several Ad-Networks were developed to compete in different styles under the specified rules of the game. But as much as we tried we did not have a full coverage of the cases agents can interact with the system and so we released the game to participating groups ahead of the planned annual competition. In exchange they could develop their own agents to compete in it and help us in the development process by interacting with the server and verifying that the results they get are consistent with the specifications.

This iterative process with the remote groups helped us stabilize the server and as time progressed we launched the first competition's qualification round where all of the agents connected simultaneously for the first time and played against each other. While this was a great milestone to achieve it did raise some issues we had to face before the actual competition that was scheduled to follow. Under the load of the different agents and messages they generated we noticed that our internal communication service jammed with messages and the simulation did not process all of the events in the allocated time frame, 10 seconds for each day. We wanted to implement a fast solution while all of the teams were still available and able to connect simultaneously and so we replaced the internal communications service from TAC's to Google's implementation and the congestion that slowed the simulation resolved.

running the research

To asses the three approaches offered for reserve price selection some minor modifications were introduced to the TAC-ADX code base in order to control which algorithm and parameters are used and export previously hidden data that was kept internally.

Static reserve price evaluation introduced a new mechanism to the platform to allow automation of the per-simulation selected reserve price. For every simulation a different static reserve price was selected, from a pre-defined range, and agents competed against each other with this value.

Dynamic reserve price adjustment involved exploring multiple configurations where the initial baseline, the update coefficient, and the radius were changed.

The DC approach required tweaking of the timing mechanism of the game to incorporate the calculation time taking place at the 10th day for a period of several minutes. With the help of *Takehiro Oyakawa* and *Amy Greenwald* we implemented a reserve price selection module based on [7] and fitted it to our ADX model. The quadratic programming was done using IBM's *CPLEX* optimizer.

Each simulation exported every auction the AdX ran. In it there was data about the query, e.g, user classification, publisher, bids sent, and the reserve price used for this specific auction. The results were later normalized to calculate average daily revenue for each approach and aggregated to evaluate overall performance. Each method was tested multiple times to account for noise generated in the simulations and validation that its performance is consistent.

Chapter 5

Workshops

5.1 Tel-Aviv University Workshop

The first TAC-ADX workshop course was taught in Tel Aviv University to under-grad C.S. students as a yearly course in 2014 in the domain of machine learning and competing agents. The students were introduced to machine learning models and developed trading agents to compete in the TAC-ADX competition. The workshop was built from weekly lectures on machine learning topics to introduce students to the domain and programming assignments targeted at gradually building a functioning and competitive agent.

In the first programming task students were asked to build a simple agent that connects to the main server, acquires a campaign, and completes it successfully. This task was built to make sure students understand how the game's messaging mechanism works, how bids are sent from the agent and reported back from the server, and how to interact with multiple server entities simultaneously. An accompanying online forum was given to the students to ask questions regarding the specifications and implementation and get updates regarding the server and agents code. Using the online moderation tool we resolved problems quickly whenever inconsistencies with the implementation and the specification aroused or when new features were in introduced to the game.

The second part of the workshop allowed the students to freely implement TAC-ADX agents according to how they perceive the different stages and aspects of the game as long as the agents were competitive. Their goal was to get the highest score when competing against their colleagues, and doing so in a fair manner - not directly interrupting with other players game strategies by exploiting special cases in the game mechanics. Using feedback from the first stage the game's specification was revised to address issues that we noticed only during run time. Some of these included: a minimum reserve price for campaign budget, to preclude agents winning campaigns by bidding very low budget and hurting all other agents, random campaign allocation where a campaign is allocated randomly to one of the bidding agents, instead of the one offering the lowest budget, to give an opportunity for agents with low quality rating to recover, and extra statistics regarding publisher impressions to give agents a better representation of the simulation.

Two more yearly workshops followed in the next years. In these workshops students had a better reference to how agents should behave and compete since the first workshop agents were available for them to train against. Some changes were made to the game's specifications before the workshops began to reflect lessons learned from previous competitions and new game mechanism were introduced to extend the game's complexity and interest.

Agent analysis - Giza

Our example workshop agent is Giza which won second place in the first international TAC-ADX competition. The agent was designed by a team of four students from Tel Aviv University, Israel.

The agent splits the game progress into three periods: (1) the beginning, first 7 days, where every agent is allocated at least one campaign and competition over impressions is harsh (2) a middle part, days 8-45, where the focus of the Giza agent is on maintaining a high quality score to stay in the game (3) a final stage, days 46-60, where the agent attempts to maximize revenue against the other agents that are left in the game.

Contract bidding strategies are chosen from a pool of multiple strategies depending on the stage of the game. In the early days there are two strategies being used intertwined: bidding a minimal price for long lasting campaigns and bidding maximal price in hopes to achieve a random allocation. The first strategy gives the agent a better chance of acquiring any campaign while the latter, if successful, allows the agent to raise its balance towards the second stage of the competition. Once the game starts its second stage the restriction on the campaign lengths for bidding the minimal price is removed and a new cost-efficient strategy is introduced. This strategy takes into account the expected revenue from each campaign, $budget - expense$, and bids only if the campaign is deemed profitable. The last stage removes the cost efficient strategy and keeps only the minimal and maximal price bidding strategies.

Figure 5.1: Bid strategies used throughout the different game stages.

Ad impression bids are generated by two strategies throughout the game. A cost-efficient strategy which predicts a bid price for each target segment and campaign, and an aggressive strategy that seeks to complete a campaign regardless to the agent profiting from it or not. The aggressive strategy is employed in the first stage of the game for every campaign in order to maintain a high quality score. Once the second stage begins the aggressive strategy handles only high budget campaigns where there is enough wiggle room for high bids, completing the campaign and gaining profit. Other campaigns are handled by the standard cost-efficient strategy in a regular fashion to balance campaign completion rate with income.

This agent has proven to be a tough competitor in the following years and have outperformed other groups who were given a chance to train against it and study it.

5.2 International Workshop

An international annual competition was held under TAC. The first took place in the *AAMAS-14*, Autonomous Agents and Multiagent Systems, conference and with the help of the TAC project. Groups from Shanghai Jiao Tong University, Technical University of Crete, Brown University, University of Edinburgh and Tel-Aviv University participated in the competition throughout the years and brought new ideas and concepts about how to trade in this complex platform.

The international competition had two stages: a qualification stage and a finals stage. Teams were required to have a running agent to compete in the qualifications and validate that all agents can interact with the main server and do not cause and disturbance in the standard flow of the game. The finals took place usually two weeks after the qualification round and agents were matched one against the other for two days. Each day had 40 consecutive simulations, each for 10 minutes, in which agents competed to get the overall best rating. Some participants published papers detailing their agent methodologies and strategies - [12] and [4]. A second competition was held at *AAMAS-15* which introduced new mechanism for reserve price selection and the third international competition was held during the 25th International Joint Conference on Artificial Intelligence - *IJCAI-16*.

Agent analysis - ANL

We will explore the winner of the first international TAC-ADX competition which took place in 2014. The agent was designed by a team from Shanghai Jiao Tong University, China [12] and outperformed its competition runner up by more than 50% in total revenue.

Two indicators are used as a basis for the agents bidding mechanisms - *Price Index (PI)* indicating segment impression cost for the upcoming trade day or a duration of several days and a *Competing Index(CI)* indicating competitiveness level of the agent. The agent employs three leading strategies: (1) it seeks to achieve every contract that is deemed profitable, i.e, can be fully completed in alloted time span and budget (2) it wishes to maintain a high quality score, as can be seen in Figure 5.2, to lower budget prices required for successful campaign allocation (3) it uses the random campaign allocation feature to acquire campaigns that would otherwise be deemed

Agent	Day 1	Day 2	Day 3	Total
ANL	1098.4	704.0	1266.5	3068.9
giza	447.8	748.6	732.4	1928.8
Agent2	596.8	454.5	363.6	1414.8
tau	-93.8	488.0	727.7	1122.0
livadx	233.5	227.7	123.4	584.5
blue	60.1	169.8	82.3	312.2
WinnieTheBot	-0.8	119.5	80.6	199.4
Amunra	-54.0	-1.8	0.0	-55.9

Table 5.1: TAC-ADX 2014 finals results

unprofitable, by bidding as high as possible.

Contract budget bids are calculated using input from the *CI* and *PI*. As the game progresses *CI* (1) remains unchanged for randomly allocated campaigns (2) increases if a campaign, deemed profitable, is not allocated to the agent (3) decreases when a campaign is allocated by a constant factor G_{greed}. As longs as the agent's quality score is above a predetermined threshold (0.8) the agent will bid the calculated price if the campaign budget is higher than the expected expense and the campaign is expected to be completed in time. If the agent's quality score is lower than the threshold it will bid the lowest possible price, as defined in the game's specifications, in hopes to secure the contract, completing it and raising the agent's quality score. If the agent's quality score is above the threshold but the campaign's budget is expected to be lower than the expected expense the agent will bid highest possible budget in hopes to get a random campaign allocation.

Price Index uses data from currently active campaigns, publishers' reserve prices, agent's quality score, and *UCS* classification level to determine bid values for each segment. The agent's strategy with regard to campaign completion is to keep a high quality score at the cost of losing profit at the end of a campaign, since quality score affects the agent's future actions, possibly for a long duration. This leads the agent to raise bid values with regard to output generated by *PI* as a contract's deadline is approaching.

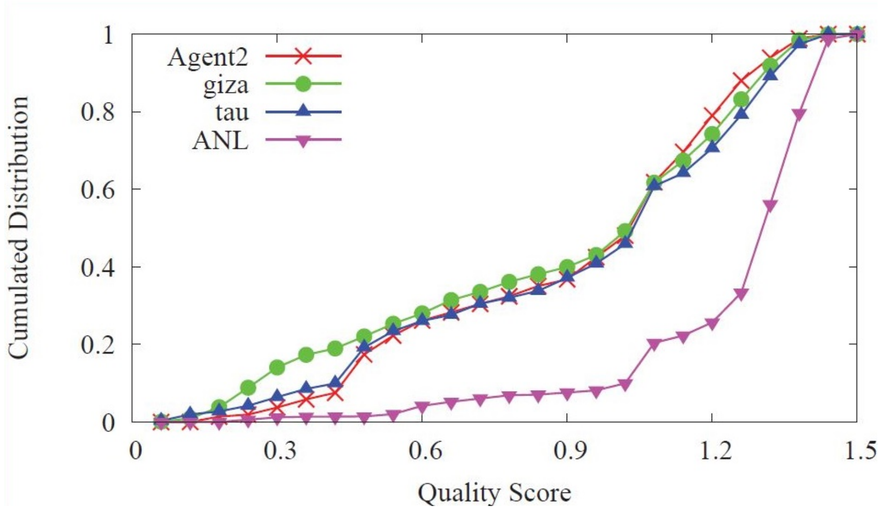

Figure 5.2: Accumulated quality score distribution for agents [12].

Bibliography

[1] Raghu Arunachalam and Norman M. Sadeh. The supply chain trading agent competition. *Electronic Commerce Research and Applications*, 4(1):66–84, 2005. 3

[2] Michael Ostrovsky Benjamin Edelman and Michael Schwarz. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *The American economic review*, 97(1):242–259, 2007. 15

[3] Nicolò Cesa-Bianchi, Claudio Gentile, and Yishay Mansour. Regret minimization for reserve prices in second-price auctions. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2013, New Orleans, Louisiana, USA, January 6-8, 2013*, pages 1190–1204, 2013. 18

[4] Moataz Elmasry. A study of trading agents for tac ad exchange. 4, 39

[5] Amy Greenwald, Takehiro Oyakawa, and Vasilis Syrgkanis. Optimal auctions with convex perceived payments. *CoRR*, abs/1601.07163, 2016. 18

[6] Patrick R. Jordan and Michael P. Wellman. Designing an ad auctions game for the trading agent competition. In *Agent-Mediated Electronic Commerce. Designing Trading Strategies and Mechanisms for Electronic Markets - AAMAS Workshop, AMEC 2009, Budapest, Hungary, May 12, 2009, and IJCAI Workshop, TADA 2009, Pasadena, CA, USA, July 13, 2009, Selected and Revised Papers*, pages 147–162, 2009. 3

[7] Mehryar Mohri and Andres Muñoz Medina. Learning theory and algorithms for revenue optimization in second price auctions with reserve. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pages 262–270, 2014. 18, 36

[8] S Muthukrishnan. Adx: a model for ad exchanges. *ACM SIGecom Exchanges*, 8(2):9, 2009. 6

[9] Jinzhong Niu, Kai Cai, Simon Parsons, Enrico Gerding, and Peter McBurney. Characterizing effective auction mechanisms: Insights from the 2007 tac market design competition. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2*, pages 1079–1086. International Foundation for Autonomous Agents and Multiagent Systems, 2008. 3

[10] Michael Ostrovsky and Michael Schwarz. Reserve prices in internet advertising auctions: a field experiment. In *Proceedings 12th ACM Conference on Electronic Commerce (EC-2011), San Jose, CA, USA, June 5-9, 2011*, pages 59–60, 2011. 18

[11] Mariano Schain and Yishay Mansour. *Ad Exchange – Proposal for a New Trading Agent Competition Game*, pages 133–145. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. 6, 18, 22, 35

[12] Bingyang Tao, Fan Wu, and Guihai Chen. Tac AdX'14: Autonomous agents for realtime ad exchange. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '15, pages 1111–1119, Richland, SC, 2015. International Foundation for Autonomous Agents and Multiagent Systems. 4, 34, 39, 41

[13] Pham Dinh Tao and Le Thi Hoai An. Convex analysis approach to dc programming: Theory, algorithms and applications. *Acta Mathematica Vietnamica*, 22(1):289–355, 1997. 18

[14] Hal R. Varian. Position auction. *International Journal of Industrial Organization*, 2006. 15

[15] Michael P. Wellman, Peter R. Wurman, Kevin O'Malley, Roshan Bangera, Shou-De Lin, Daniel M. Reeves, and William E. Walsh. Designing the market game for a trading agent competition. *IEEE Internet Computing*, 5(2):43–51, 2001. 3

תקציר

עולם הפרסום עובר מהפיכה בעשרים השנים האחרונות ובו עיקר הפרסום נודד לפרסום המקוון. עולם זה מכיל את אתרי האינטרנט (הן במחשבים והן במכשירים הניידים), אפליקציות במכשירים ניידים, וממשקים נוספים בעולם הדיגיטלי שסביבנו. החלק הארי בעולם פרסום זה הינם פרסומות המופיעות במסגרת תוצאות חיפוש ממומנות (sponsored ads) ושטחי פרסום (banner and video ads).

שטחי הפרסום מהווים כיום ערוץ הכנסה משמעותי למגוון רחב של חברות בעולם האינטרנט. כך בשנת 2017, לדוגמא, סך ההכנסות משטחי פרסום הגיע ל- 200 מילארד דולר בעולם, ול- 70 מיליארד דולר בארצות הברית. עיקר ההכנסות בארצות הברית זרמו לשתי חברות מרכזיות: גוגל (Google) ופייסבוק (Facebook), כאשר עבור פייסבוק הפרסום מהווה ערוץ הכנסות מרכזי, כ-95% מסך הכנסותיה.

הפרסום הדיגיטלי בשטחי הפרסום דורש קישור של החברות המסחריות (המעוניינות לפרסם) לאתרי האינטרנט (המשכירים את שטחי הפרסום). בתחילת הדרך הקישור התבצע בצורה ישירה בין החברה המסחרית לאתר האינטרנט. חיבור זה יצר תלות של אתרי האינטרנט במפרסמים איתם יצרו קשר וגם לחוסר גמישות ביכולתם לבחור מבין מספר מפרסמים. בהמשך התפתח המודל ונוספו לו חברות פרסום אשר שכרו שטחי פרסום באתרים השונים, ומכרו אותם לחברות המסחריות בסיטונאות. בשיטה זו יכלו אתרי האינטרנט לממש שטחי פרסום אותם לא מכרו ישירות למפרסמים בעלות מופחתת. לבסוף התפתח המודל לכדי המצב הנוכחי בו ישנן בורסות פרסום (Ad Exchanges) אשר מקשרות בין אתרים למפרסמים ולסוכנויות פרסום בצורה יעילה ונוחה לשני הצדדים.

בורסת פרסום הינה שיטה ממוחשבת המאפשרת חיבוריות בין אתרי האינטרנט לחברות מסחריות בצורה מקוונת, מהירה ונוחה. במסגרת שיטה זו מעדכן אתר האינטרנט את הבורסה על שטחי פרסום שאמורים להיות מוצגים לגולש אינטרנט, וסוכנויות הפרסום משתתפות במכרז GSP (Generalized Second Price) באם ברצונם להציג פרסומת לגולש זה. כך לדוגמא: משתמש גולש לאתר ynet באמצעות המחשב או האפליקציה, ynet מודיעה לבורסה (שירות ה-AdExchange של גוגל) על הזדמנות להציג פרסומת בדפיה, וכן מעדכנת פרטים כלליים על המשתמש הגולש. רשתות הפרסום, המייצגות את החברות המסחריות, מציעות את תוכן הפרסום והיקף התשלום האפשרי וגוגל, באמצעות מכרז, בוחרת את ההצעה המתאימה ביותר ומחזירה לאתר את הפרסומת אשר תוצג לגולש.

השימוש בבורסות פרסום מקוונת דורש מהשיטה להיות מהירה מאד, מגיבה בזמן-אמת ואפקטיבית לכל המשתתפים. האפקטיביות נמדדת בראש ובראשונה בהיקף התשלום שהחברות המסחריות נדרשות לשלם, והיקף ההכנסה לאתרי הפרסום. היקף תשלום זה נקבע על ידי הגדרת גובה הצעה מינימלית על ידי האתר המפרסם, ועל ידי גובה התשלום של החברה המפרסמת, הנקבע באמצעות מכרז המבוצע בזמן אמת בין כל החברות.

אוסטרובסקי ושות' [1] מציגים במאמרם ניתוח של מכרזי GSP אשר נמצאים בליבה של מערכת המכרזים של בורוסת המסחר האינטרנטיות. הם מראים כי להבדיל ממכרזי VCG (Vickrey-Clarke-Groves) במכרזי GSP הצעת מחיר האמת בו מוערכת הסחורה אינה אסטרטגיה דומיננטית למרות הדמיון בין שני סוגי המכרזים. וריאן [2] מציג במחקרו ניתוח של מודל המדמה את שוק מכרזי הפרסומות שבשימוש גוגל ויאהו.

הרציונאל המרכזי של המחקרים שבוצעו עד כה היה נסיון להבנות מודל מתימטי שידמה את התנהגות השחקנים השונים במכרז, ובצורה זו להעריך את צורת התנהגותם במשחק ולהציע כיצד עליהם להתנהג בשוק זה על מנת לשפר את הכנסותיהם ואת המדדים עליהם הם נמדדים, בין אם אלו סוכנים אשר מייצגים סוכנויות פרסום או אתרי אינטרנט המנסים למקסם את רווחיהם במכרזים. האתגר המרכזי בשיטה זו היה להבין האם המודל תקף גם בזמן שהוא רץ בפועל בשוק והשחקנים האחרים מגיבים אליו. לצורך כך הועתק בשנים האחרונות המודל של משחק בין "סוכנים חכמים" מקוונים (Smart Agents) אשר מאפשר לדמות את הפעילות של בורסות הפרסום. במודל משחק זה כל "סוכן חכם" יכול לדמות אסטרטגיה של אתר פרסום או של חברת מסחרית, וכך ללמוד טוב יותר על ההתנהגות העסקית הנדרשת.

מחקר יישומי באינטיליגנציה מלאכותית בכלל, ובמערכות מרובות סוכנים בפרט עסק בתחום זה. מטרת העל הייתה לפתח יכולת להשוות בצורה אמפירית שיטות שונות לסחר. תחרות הסוכנים הסוחרים (Trading Agents Competition) הציעה מודל סטנדרטי אשר אפשר לקבוצות מרחבי העולם לפתח סוכנים להתחרות כנגד אחרים בסביבת מסחר דיגיטלית. הסביבה ניסתה למדל דוגמאות מהחיים בהן מספר סחורות היו קשורות זו בזו ואסטרטגיות למסחר בהן היו צריכות להתחשב ביחסים בינהן. התחרות הראשונה מסוג זה נוהלה בשנת 2000 בכנס ICMAS עם כ20 קבוצות מתחרות מרחבי העולם.

משחק ה TAC הראשון, [3], עסק בסוכני נסיעות ובתפקידם לארגן טיולים ללקוחותיהם. כל סוכן מקבל לאורך המשחק משימות מלקוחות המבקשים לטייל ועליו לארגן לכל לקוח מסלול שלם הכולל הזמנת טיסות, בית מלון ואירועי בידור. הסוכן אחראי לזמן את שלוש הסחורות הללו עבור כל לקוח בהתאם לבקשותיו ובתקציב העומד לרשותו ומדדי ההצלחה נסמכים על ביצועיו של הסוכן במשימות אלו.

לאורך השנים גרסאות שונות של תחרות הסוכנים החכמים הוצגו כאשר כל אחת עסקה בשוק סחורות שונה. גרסת שרשרת האספקה TAC-SCM עסקה ברכש סחורות עבור קו אספקה. כאשר בקשה הגיעה ליצרן עליו היה ליצור קשר עם ספקי החלקים השונים הדרושים להרכבת המוצר, לרכוש את הסחורה המתאימה ולחייב את הלקוח בהתאם. בין הסחורות השונות היו קשרים כך שרכישת מוצר מסויים חייבה רכישתו של אחר על מנת שיתאימו אחד לשני בהליך ההרכבה.

גרסה נוספת של תחרות הסוכנים החכמים, [4], הציגה מנגנון שונה בו המוכרים והקונים בשוק ממושו על ידי מפתחי המשחק. על המשתתפים היה לחבר בין הסוחרים לקונים שרצו לסחור זה עם זה ע"י התאמת ההיצע לדרישה וקישור

בין השניים. הסוכנים יכלו לגבות תשלום מהסוחרים על מנת שייצגו אותם בתהליך, עמלה על כל עסקה העוברת דרכם, עמלה על הצגת מידע על מצב השוק ועוד.

הגרסה הרביעית של המשחק הציגה לעולם הסוכנים החכמים את עולם הפרסום. תחרות TAC-AA (Trading Agents Competition - Ad Auction) [5] התמקדה בפעילות הסוכנים בסביבת פרסומות חיפוש ממומנות, וביחסים בין רשתות פרסום, אתרים ומשתמשים. בסביבה זו על המשתתפים לממש סוכן, המייצג מפרסם המעוניין לקדם את מוצרו, ולהתחרות כנגד מפרסמים אחרים על הצגת פרסומות למשתמשים הצמודות לחיפושים במנועי חיפוש. הסוכנים נמדדו על פי יכולתם למשוך מבקרים באתרי החיפוש לאתריהם והמרת מעבר זה לכדי רכישת סחורה בפועל.

במקביל לתחרויות ה- TAC התקיימו תחרויות ונספות. משמעותית שבהן היתה תחרות Pay Per Click Bidding Agent Competition שנוהלה בכנס ACM EC-06 ואפשרה למשתתפים לנהל קמפיינים אמיתיים לפרסום מונחה מילות חיפוש תחת מרכז הפרסומות של מייקרוסופט (Microsoft AdCenter). ייחודה של תחרות זו היה בכך שבפעם הראשונה יכלו קבוצות מרחבי העולם לבחון בצורה מבוקרת ובמודל מעשי אסטרטגיות לניהול סוכנים ממחושבים המתחרים במכרזי פרסומות מקוונים.

פרסום מותגים אשר בו המפרסמים לא נמדדים בהכרח בכמות הסחורה הנמכרת, אלא בהיקף תנועה וחשיפת משתמשים למסרים, לא מתאים למודל הקיים של TAC-AA. בנוסף, נוכחותה הבולטת של בורסת הפרסום אשר מנהלת חלק נכבד מהתקשורת בין רשתות הפרסום והאתרים השונים לא באה לידי ביטוי במודל הקיים אשר מניח כי קיימת תקשורת ישירה בין המפרסמים לאתרים. המשחק החדש אותו אנו מציגים מוסיף גם את רשתות הפרסום אשר אחראיות לתווך מספר מפרסמים אל עולם הפרסום וגם את בורסת הפרסומות (AdExchange) אשר מנהלת את כל התקשורת בין סוכנויות הפרסום לאתרים. המשתתפים בתחרויות הTAC-ADX נדרשים לבנות סוכן המייצג רשת פרסום ועליהם להציג פרסומות לגולשים באתרים על פי דרישות המפרסמים. בצורה זו אנו מאפשרים לחוקרים לבחון כיצד אסטרטגיות שונות לניהול מכרזים משפיעות על מאזן הכוחות בשוק ומתפקדות בפועל כנד מתחרים אמתיים המגיבים לצעדיהם בזמן אמת.

בעבודה זו אנו מציגים מודל חדש לתחרות TAC המאפשר מידול של פרסום מותגים ובחינת אסטרטגיות פרסום שונות. מטרת המודל החדש הינה לאפשר לחקור לאורך זמן כמות גדולה של סוכנים אשר מממשים אסטרטגיות ניהול קמפיינים ומחקר רכיבים שונים בשוק הפרסום. יכולתו של המודל לדמות את העולם האמיתי הביאה לכך שהמודל המוצג במחקר זה היווה תשתית מחקר ומשחק בעולם הסוכנים החכמים לפרסום, [7] ,[6]. בנוסף, על בסיס המודל החדש בצענו מחקר של השפעת מחיר המינימום של אתרי האינטרנט במכרזים על הרווחים שלהם. המחקר מציג את הקשר בין מחיר המינימום לגובה ההצעות וסך ההכנסות של האתרים לאורך זמן.

בעבודה זו ארבעה פרקים. בפרק הראשון מוצג עולם הבעייה ומרחב הפתרונות שהיה קיים עד כה. בפרק השני מתואר בפירוט מודל ה-Ad-Exchange על חלקיו השונים, ומתואר המחקר והיישום שבוצעו על מנת להתאימו לסביבת

TAC. במסגרת המודל הכללי מוצגים השחקנים העיקריים בשוק מסחר הפרסומות המקוון – משתמשים, מפרסמים, רשתות הפרסום ובורסת הפרסומות. במודל ה-TAC מוצגים רעיונות ושירותים בעולם הפרסום המקוון הקשורים למשחק ובהם שירותי סיווג משתמשים, קמפייני פרסום ועוד.

בפרק השלישי בוצע מחקר של השפעת מחיר המינימום על חברות האינטרנט. במסגרת מחקר זה השווינו בין מספר שיטות לקביעת מחירי מינימום עבור מכרזים לפרסומות – שיטה סטטית, שיטה דינמית ושיטה המתבססת על הפרשי פונקציות קמורות (Difference of Convex). הראנו כי השיטה האחרונה הייתה היעילה ביותר לאורך זמן.

בפרק הרביעי אנו מציגים את מבנה המערכת שהוקמה לצורך המחקר – TAC-ADX. את האתגרים בכתיבתה ובהרצת התחרות עם משתתפים אשר מתחרים בה מרחוק, את הסדנאות שנוהלו באוניברסיטת תל אביב ואת התחרויות הבינלאומיות שלשמן נכתב המודל והמשחק מלכתחילה.

[1] Michael Ostrovsky Benjamin Edelman and Michael Schwarz. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. The American economic review, 97(1):242{259, 2007.

[2] Hal R. Varian. Position auction. International Journal of Industrial Organization, 2006.

[3] Michael P. Wellman, Peter R. Wurman, Kevin O'Malley, Roshan Bangera, Shou-De Lin, Daniel M. Reeves, and William E. Walsh. Designing the market game for a trading agent competition. IEEE Internet Computing, 5(2):43{51, 2001.

[4] Jinzhong Niu, Kai Cai, Simon Parsons, Enrico Gerding, and Peter McBurney. Characterizing e_ective auction mechanisms: Insights from the 2007 tac market design competition. In Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2, pages 1079{1086. International Foundation for Autonomous Agents and Multiagent Systems, 2008.

[5] Patrick R. Jordan and Michael P. Wellman. Designing an ad auctions game for the trading agent competition. In Agent-Mediated Electronic Commerce. Designing Trading Strategies and Mechanisms for Electronic Markets - AAMAS Workshop, AMEC 2009, Budapest, Hungary, May 12, 2009, and IJCAI Workshop, TADA 2009, Pasadena, CA, USA, July 13, 2009, Selected and Revised Papers, pages 147–162, 2009

[6] Bingyang Tao, Fan Wu, and Guihai Chen. Tac adx’14: Autonomous agents for realtime ad exchange. In Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, AAMAS ’15, pages 1111–1119, Richland, SC, 2015. International Foundation for Autonomous Agents and Multiagent Systems.

[7] Moataz Elmasry. A study of trading agents for tac ad exchange.

TEL AVIV UNIVERSITY אוניברסיטת תל-אביב
הפקולטה למדעים מדויקים ע"ש ריימונד ובברלי סאקלר
בית הספר למדעי המחשב ע"ש בלבטניק

משחק בורסת הפרסומות

חיבור זה הוגש כחלק מהדרישות לקבלת תואר

"מוסמך אוניברסיטה" – .M.Sc

באוניברסיטת תל-אביב
ביה"ס למדעי המחשב ע"ש בלבטניק

על ידי

תומר גרינוולד

העבודה הוכנה בהנחיית

פרופ' ישי מנצור

פברואר 2018