Complex Models

- **Simulated Markets**
  - Trading Agent Competition (TAC)
    - TAC Games Overview
    - TAC Ad Auctions Game Details
    - Ad Auction Agent Examples
  - JACK: in-the-box Java Auction Configuration Kit

- **Real Markets**
  - QuiBids: A penny auction site and a trading agent
Trading Agent Competition (TAC)

http://aa.tradingagents.org
Trading Agent Research

- **Goal:** to develop techniques for the effective design and analysis of trading agents
  - both specific solutions to particular trading problems and general principles to guide the development of trading agents

- **Challenge:** markets are multiagent environments where an agent’s performance depends on other agents’ behavior
  - natural approach is for separate institutions/researchers to develop agents to participate in a common market environment
Trading Agent Research

• **Goal:** to develop techniques for the effective design and analysis of trading agents
  
  • both specific solutions to particular trading problems and general principles to guide the development of trading agents

• **Challenge:** markets are multiagent environments where an agent’s performance depends on other agents’ behavior
  
  • natural approach is for separate institutions/researchers to develop agents to participate in a common market environment

Enter Trading Agent Competition (TAC) ...
Trading Agent Research

• **Goal:** to develop techniques for the effective design and analysis of trading agents

- both specific solutions to particular trading problems and general principles to guide the development of trading agents

• **Challenge:** markets are multiagent environments where an agent’s performance depends on other agents’ behavior

- natural approach is for separate institutions/researchers to develop agents to participate in a common market environment

Enter Trading Agent Competition (TAC) ...

Since 2000, teams from: US, UK, Sweden, Greece, Turkey, Croatia, Hong Kong, Australia, Israel, ...
TAC Games
TAC Games

Travel
TAC Games

Travel

SCM
TAC Games

Travel

SCM

MARKET BASED CONTROL
OF COMPLEX COMPUTATIONAL SYSTEMS
TAC Games

Travel

SCM

CAT

AA
TAC Games

Travel

Power

CAT

SCM

Ad Auctions

AA
TAC Goals

Stylized Models  TAC  Real World

Don’t usually capture real-world complexity

Don’t want financial risk and market noise
TAC Goals

- 6-8 agents compete in each game
- Simulate many games with different market conditions
- Agent with highest average profit wins
- Experiments are replicable

Stylized Models  TAC  Real World

Don’t usually capture real-world complexity  Don’t want financial risk and market noise
TAC AA
Game Entities
Advertisers bid on keywords, select ad types and budgets.

Publisher ranks ads for each keyword.

Users issue queries, click on ads, convert (i.e., buy products).

Daily summaries disseminated.

Server: Sends advertisers private information.

Advertiser’s goal: Maximize profits.

Daily Cycle (60 days)
Market

Manufacturers
- Lioneer
- PG
- Flat

Components
- TV
- Audio
- DVD

Products
Queries

“PG TV”
Queries

“TV”

Semi-Targeted Queries

Targeted Queries
Queries

“Electronics”

Generic Query

Semi-Targeted Queries

Targeted Queries
How do users place queries, click, and convert on ads?
User Query Model

- Searching Users
  - Not Searching
  - Info Search
  - Generic Search
  - Semi-Targeted Search
  - Targeted Search
  - Transacted
## Cascade Model

<table>
<thead>
<tr>
<th>Slot</th>
<th>Advertiser</th>
<th>Ad Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>Generic</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

"TV."

### Click? | Convert? | Continue?
--- | --- | ---
advertiser quality, ad type | user state, capacity, component specialty | user state

<table>
<thead>
<tr>
<th>Event</th>
<th>Pr(Event)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click</td>
<td>advertiser quality, ad type</td>
</tr>
<tr>
<td>Convert</td>
<td>user state, capacity, component specialty</td>
</tr>
<tr>
<td>Continue</td>
<td>user state</td>
</tr>
</tbody>
</table>
## Cascade Model

<table>
<thead>
<tr>
<th>Slot</th>
<th>Advertiser</th>
<th>Ad Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Generic</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Lioner</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Google</td>
<td></td>
</tr>
</tbody>
</table>

User thinks: “TV.”

Click? N, Convert? N, Continue? Y

Query
## Cascade Model

<table>
<thead>
<tr>
<th>Slot</th>
<th>Advertiser</th>
<th>Ad Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Generic</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>User</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>User</td>
<td></td>
</tr>
</tbody>
</table>

User queries are processed in a cascade model, with each slot representing a step in the process. The table shows the sequence of ad types, with 'Generic' being the initial ad type followed by specific user queries.

The user thinks of the query as "TV."
## Cascade Model

<table>
<thead>
<tr>
<th>Slot</th>
<th>Advertiser</th>
<th>Ad Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Generic</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

User queries "TV." and considers clicking, converting, or continuing.

---

**Query**
# Cascade Model

<table>
<thead>
<tr>
<th>Slot</th>
<th>Advertiser</th>
<th>Ad Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>Generic</td>
</tr>
<tr>
<td>2</td>
<td>Lioneer</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

User queries: “TV.”

- Click? Y, Convert?, Continue?
# Cascade Model

<table>
<thead>
<tr>
<th>Slot</th>
<th>Advertiser</th>
<th>Ad Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Blue</td>
<td>Generic</td>
</tr>
<tr>
<td>2</td>
<td>Green</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Yellow</td>
<td></td>
</tr>
</tbody>
</table>

User:
- Query: “TV.”
- Click? Y, Convert?, Continue?

Landing Page:
- "TV" in search bar.
- Screen with search results for TVs.
## Cascade Model

<table>
<thead>
<tr>
<th>Slot</th>
<th>Advertiser</th>
<th>Ad Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Generic</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

User sees “TV.”

Click? Y, Convert? N, Continue? Y
<table>
<thead>
<tr>
<th>Slot</th>
<th>Advertiser</th>
<th>Ad Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>Generic</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Cascade Model**

- **User**
  - Click?, Convert?, Continue?
  - "TV."

... until a conversion or no more ads
How does publisher position and price ads?
<table>
<thead>
<tr>
<th>Advertiser</th>
<th>Bid</th>
<th>Quality</th>
<th>Score</th>
<th>Slot</th>
<th>CPC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Advertiser</td>
<td>Bid</td>
<td>Quality</td>
<td>Score</td>
<td>Slot</td>
<td>CPC</td>
</tr>
<tr>
<td>------------</td>
<td>------</td>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td>$4</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$3</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$2</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$1</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Advertiser</td>
<td>Bid</td>
<td>Quality</td>
<td>Score</td>
<td>Slot</td>
<td>CPC</td>
</tr>
<tr>
<td>------------</td>
<td>-----</td>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td>$4</td>
<td>1</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$3</td>
<td>3</td>
<td>9</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$2</td>
<td>6</td>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$1</td>
<td>8</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Advertiser</td>
<td>Bid</td>
<td>Quality</td>
<td>Score</td>
<td>Slot</td>
<td>CPC</td>
</tr>
<tr>
<td>------------</td>
<td>-----</td>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td>$4</td>
<td>1</td>
<td>4</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$3</td>
<td>3</td>
<td>9</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$2</td>
<td>6</td>
<td>12</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$1</td>
<td>8</td>
<td>8</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Advertiser</td>
<td>Bid</td>
<td>Quality</td>
<td>Score</td>
<td>Slot</td>
<td>CPC</td>
</tr>
<tr>
<td>------------</td>
<td>-----</td>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td>$4</td>
<td>1</td>
<td>4</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$3</td>
<td>3</td>
<td>9</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$2</td>
<td>6</td>
<td>12</td>
<td>1</td>
<td>(9/6) = $1.50</td>
</tr>
<tr>
<td></td>
<td>$1</td>
<td>8</td>
<td>8</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Advertiser</td>
<td>Bid</td>
<td>Quality</td>
<td>Score</td>
<td>Slot</td>
<td>CPC</td>
</tr>
<tr>
<td>------------</td>
<td>-----</td>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td></td>
<td>$4</td>
<td>1</td>
<td>4</td>
<td>X</td>
<td>(8/3) = $2.67</td>
</tr>
<tr>
<td></td>
<td>$3</td>
<td>3</td>
<td>9</td>
<td>2</td>
<td>(9/6) = $1.50</td>
</tr>
<tr>
<td></td>
<td>$2</td>
<td>6</td>
<td>12</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$1</td>
<td>8</td>
<td>8</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

Minimum bid to retain slot
<table>
<thead>
<tr>
<th>Advertiser</th>
<th>Bid</th>
<th>Quality</th>
<th>Score</th>
<th>Slot</th>
<th>CPC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$4</td>
<td>1</td>
<td>4</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>$3</td>
<td>3</td>
<td>9</td>
<td>2</td>
<td></td>
<td>(8/3) = $2.67</td>
</tr>
<tr>
<td>$2</td>
<td>6</td>
<td>12</td>
<td>1</td>
<td></td>
<td>(9/6) = $1.50</td>
</tr>
<tr>
<td>$1</td>
<td>8</td>
<td>8</td>
<td>3</td>
<td></td>
<td>(4/8) = $0.50</td>
</tr>
<tr>
<td>Advertiser</td>
<td>Bid</td>
<td>Quality</td>
<td>Score</td>
<td>Slot</td>
<td>CPC</td>
</tr>
<tr>
<td>------------</td>
<td>-----</td>
<td>---------</td>
<td>-------</td>
<td>------</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td>$4</td>
<td>1</td>
<td>4</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>$3</td>
<td>3</td>
<td>9</td>
<td>2</td>
<td>(8/3) = $2.67</td>
</tr>
<tr>
<td></td>
<td>$2</td>
<td>6</td>
<td>12</td>
<td>1</td>
<td>(9/6) = $1.50</td>
</tr>
<tr>
<td></td>
<td>$1</td>
<td>8</td>
<td>8</td>
<td>3</td>
<td>(4/8) = $0.50</td>
</tr>
</tbody>
</table>

Minimum bid to retain slot
### Minimum bid to retain slot

<table>
<thead>
<tr>
<th>Advertiser</th>
<th>Bid</th>
<th>Quality</th>
<th>Score</th>
<th>Slot</th>
<th>CPC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$4</td>
<td>1</td>
<td>4</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>$3</td>
<td>3</td>
<td>9</td>
<td>2</td>
<td>(8/3) = $2.67</td>
</tr>
<tr>
<td></td>
<td>$2</td>
<td>6</td>
<td>12</td>
<td>1</td>
<td>(9/6) = $1.50</td>
</tr>
<tr>
<td></td>
<td>$1</td>
<td>8</td>
<td>8</td>
<td>3</td>
<td>(4/8) = $0.50</td>
</tr>
</tbody>
</table>
What information do advertisers have and what decisions do they face?
Ad Types

Generic Ad

Targeted Ad

Semi-Targeted Ad

Generic Ad
## Query & Sales Reports

<table>
<thead>
<tr>
<th>Advertiser</th>
<th>Ad Type</th>
<th>Average Slot</th>
<th>Average CPC</th>
<th>Impressions</th>
<th>Clicks</th>
<th>Conversions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generic</td>
<td>1.0</td>
<td>1.2</td>
<td>2.57</td>
<td>1000</td>
<td>200</td>
<td>50</td>
</tr>
<tr>
<td>Generic</td>
<td>1.2</td>
<td>2.57</td>
<td></td>
<td>1000</td>
<td>200</td>
<td>50</td>
</tr>
<tr>
<td>Generic</td>
<td>2.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Advertiser Decisions

Private Information
- Manufacturer Specialization
- Component Specialization
- Distribution Capacity

For this query type:
- Bid
- Budget
- Ad type

Daily Reports
- Query Reports
- Sales Reports

Overall budget

Publisher

Advertiser
Outsourcing Creativity

Tournament Structure

Finals

Semifinals

Semifinals

Qualifiers

Qualifiers

TAC 2009 Finals

<table>
<thead>
<tr>
<th>Agent</th>
<th>Affiliation</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>TacTex</td>
<td>U Texas @ Austin</td>
<td>79.9</td>
</tr>
<tr>
<td>AstonTAC</td>
<td>Aston U</td>
<td>76.3</td>
</tr>
<tr>
<td>Schlemazl</td>
<td>Brown U</td>
<td>75.4</td>
</tr>
<tr>
<td>QuakTAC</td>
<td>U Penn</td>
<td>74.5</td>
</tr>
<tr>
<td>DNAgents</td>
<td>U Wash @ Tacoma</td>
<td>71.8</td>
</tr>
<tr>
<td>EPFLAgent</td>
<td>EPFL</td>
<td>71.7</td>
</tr>
<tr>
<td>MetroClick</td>
<td>City U of New York</td>
<td>70.6</td>
</tr>
<tr>
<td>UMTac09</td>
<td>U Macau</td>
<td>66.9</td>
</tr>
</tbody>
</table>
TacTex

PREDICT market prices

OPTIMIZE: i.e., make decisions

Position Analyzer
- impressions
- avg. positions

Estimation and Prediction
- total impressions
- bid ranks
- impression ranges

User Model
- predicted users in each state
- predicted bids
- predicted impressions
- predicted ads

Advertiser Model
- ads
- cost per click

Parameter Model
- parameter estimates

Parameter Model
- clicks

Other advertisers

Game Server
- TacTex’s daily bids, ads, and spending limits for all queries

Multi-day Optimizer

Single-day Optimizer
- Query Analyzer

Pardoe, Chakraborty, & Stone
University of Texas at Austin
TacTex

PREDICT market prices

OPTIMIZE: i.e., make decisions

TacTex

Position Analyzer

- impressions
- avg. positions

Estimation and Prediction

User Model
- total impressions
- predicted users in each state
- predicted bids
- predicted impressions
- predicted ads

Advertiser Model
- bid ranks
- predicted impressions
- predicted ads

Parameter Model
- impression ranges
- predicted impressions
- predicted ads
- parameter estimates

Other advertisers

Game Server

- TacTex’s daily bids, ads, and spending limits for all queries

Pardoe, Chakraborty, & Stone
University of Texas at Austin
Adaptive bidding based on overall target capacity, $C^*$

If today’s sales are too high, reduce tomorrow’s sales by decreasing bids.
Distribute target capacity using regret minimization

\[ c_{q}^{t+1} = \frac{e^{\eta v_{q}^{t-1}}}{\sum_{q} e^{\eta v_{q}^{t-1}}} C_{t+1} \]

**Aim:** Robust Agent

- Modeler
- Estimator
- Optimizer
QuakTac (and later Mertacor)

- Game-theoretic agent
- Bids $\alpha v$: $\alpha < 1$, $v$ is value/click
- Finds $\alpha$ using best-reply dynamics
- Restricted strategy space
- Initialized at truth-telling
**Schlemazl**

**Input:** Daily Summaries

**Output:** Bids, Budgets, Ads

- **Summary Disaggregator**
  - **User Model Estimation**
  - **Parameter Estimation**
  - **Budget Estimation**
  - **Bid Estimation**
  - **Optimization**

**Server**
**Schlemazl**

**Input:** Daily Summaries

**Output:** Bids, Budgets, Ads

**PREDICT market prices**
- User Model Estimation
- Parameter Estimation
- Summary Disaggregator

**OPTIMIZE:** i.e., make decisions
- Optimization
- Budget Estimation
- Bid Estimation

**Server**

Daily Summaries

Bids, Budgets, Ad Types
Which prediction method is best?

Which optimization routine is best?

Input: Daily Summaries

Output: Bids, Budgets, Ads
Which prediction method is best?
Which optimization routine is best?

Input: Daily Summaries

Output: Bids, Budgets, Ads
Schlemazl

Summary Disaggregator

User Model Estimation

Parameter Estimation

Budget Estimation

Optimization

Daily Summaries

Bids, Budgets, Ad Types

Server

\{1.0, 2.0, 2.2, 1.88888889, 1.46666667\}

\(N = 20\)
Aggregation Bias in Sponsored Search Data
Abhishek, Hosanagar, & Fader
Upside-down Waterfall

N = 20
N = 7
N = 5
N = 10
N = 20
N = 25

Impressions

Slots

Tuesday, July 24, 12
Upside-down Waterfall

\[ [1.0, 2.0, 2.2, 1.46666667, 1.88888889] \]

- N = 7
- N = 5
- N = 10
- N = 20
- N = 25

Impressions

Tuesday, July 24, 12
Upside-down Waterfall

1.0  N = 7
2.0  N = 5
2.2  N = 10
1.46666667  N = 20
1.88888889  N = 25

N = 20
N = 7
N = 5
N = 10
N = 20
N = 25

Impressions

Slots

Tuesday, July 24, 12
Upside-down Waterfall

\[
\begin{bmatrix}
1.0 \\
2.0 \\
2.2 \\
1.46666667 \\
1.88888889
\end{bmatrix}
\]

N = 7
N = 5
N = 10
N = 20
N = 25

Slots

Impressions

Tuesday, July 24, 12
Heuristic Search

Impressions Priors

\[
\begin{bmatrix}
1.0 \\
2.0 \\
2.2 \\
1.46666667 \\
1.88888889 \\
\end{bmatrix}
\]

\[N = 20\]

Impressions

Slots

Tuesday, July 24, 12
Evaluate Waterfall

$$\arg \max_{\mathcal{T}_i^a} \prod_{a \neq a^*} \Pr(\mathcal{T}_a = \sum_i I_i^a | \text{models})$$
Prediction Challenge

**GOAL:** to compare the performance of prediction methods, without running too many expensive game experiments

- Run a few real games and log all agents’ decisions and game outcomes
- Pick a candidate prediction method to substitute for the one used by one of the original agents
  - Send the candidate all the information the original agent received
  - Compare the candidate’s predictions with the outcome recorded in the log
- Repeat using all game logs and all agents, for all candidate prediction methods
## Prediction Challenge

<table>
<thead>
<tr>
<th>ε(I_{j,k})</th>
<th>ε(T_j)</th>
<th>ε(T)</th>
</tr>
</thead>
<tbody>
<tr>
<td>crocodileagent</td>
<td>24</td>
<td>38</td>
</tr>
<tr>
<td>TacTex</td>
<td><strong>32</strong></td>
<td><strong>79</strong></td>
</tr>
<tr>
<td>tau</td>
<td>10</td>
<td>21</td>
</tr>
<tr>
<td>MetroClick</td>
<td>24</td>
<td>41</td>
</tr>
<tr>
<td>Schlemazl</td>
<td>19</td>
<td>37</td>
</tr>
<tr>
<td>McCon</td>
<td>24</td>
<td>46</td>
</tr>
<tr>
<td>Mertacor</td>
<td><strong>32</strong></td>
<td><strong>65</strong></td>
</tr>
</tbody>
</table>

Participates in many auctions; drops out early.

Participates in few auctions; stays in until end.

### Prediction error depends on agent’s decisions.
Can’t look at predictions in isolation.
Future Work

- Measure error propagation
- Measure impact on the bottom line
Schlemazl

Optimization Challenge

Summary Disaggregator

User Model Estimation

Parameter Estimation

Budget Estimation

Bid Estimation

Optimization

Input:
Daily Summaries

Output:
Bids, Budgets, Ads

Server

Daily Summaries

Bids, Budgets, Ad Types

Tuesday, July 24, 12
Model-heavy optimization

How many sales do we anticipate?

...and how much profit?
Model-heavy optimization

How many sales do we anticipate?

...and how much profit?

Models give us these values.

\[ v_{qs} = i_q \cdot Pr\text{Click}_{qs} \cdot (Pr\text{Conv}_q \cdot USP_q - CPC_{qs}) \]

\[ w_{qs} = i_q \cdot Pr\text{Click}_{qs} \cdot Pr\text{Conv}_q \]
Model-heavy optimization

What bids should we place, given this information?

Models give us these values.

\[ w_{qs}, v_{qs} \]

| 90,20 | 95,19 |
| 89,18 | 89,18 |
| 60,10 | 60,10 |
| 20,4  | 20,4  |
| 10,3  | 10,3  |
| 67,18 |
| 65,13 |
| 60,10 |
| 20,4  |
| 10,3  |

\[
\begin{align*}
    v_{qs} &= i_q PrClick_{qs}(PrConv_q USP_q - CPC_{qs}) \\
    w_{qs} &= i_q PrClick_{qs} PrConv_q
\end{align*}
\]
Model-heavy optimization

What bids should we place, given this information?

Models give us these values.

\[
v_{qs} = i_q \cdot PrClick_{qs} \cdot (PrConv_{qs} \cdot USP_q - CPC_{qs})
\]

\[
w_{qs} = i_q \cdot PrClick_{qs} \cdot PrConv_{qs}
\]
Model-heavy optimization

Penalized

What bids should we place, given this information?

Models give us these values.

\[
\begin{align*}
    v_{qs} &= i_q \cdot \text{PrClick}_{qs} \cdot (\text{PrConv}_{qUSP_q} - \text{CPC}_{qs}) \\
    w_{qs} &= i_q \cdot \text{PrClick}_{qs} \cdot \text{PrConv}_q
\end{align*}
\]
Equimarginal Principle

H.H. Gossen

If a man is free to choose among several pleasures but has not time to afford them all to their full extent, then in order to maximize the sum of his pleasures he must engage in them all to at least some extent before enjoying the largest one fully, so that the amount of each pleasure is the same at the moment when it is stopped; and this however different the absolute magnitude of the various pleasures may be.
Equimarginal Principle

H.H. Gossen

If a man is free to choose among several pleasures but has not time to afford them all to their full extent, then in order to maximize the sum of his pleasures he must engage in them all to at least some extent before enjoying the largest one fully, so that the amount of each pleasure is the same at the moment when it is stopped; and this however different the absolute magnitude of the various pleasures may be.

Aim: Equate marginal ROI across queries
Model-heavy algorithms equate marginal ROI using heavy-duty models

Model light algorithms equate some proxy of marginal ROI using less information, such as ROI

Models give us these values.
If today’s sales are too high, reduce tomorrow’s sales by increasing target ROI, thereby decreasing bids.

**Input:** sales, targetROI, targetSales

**Output:** \( \forall q, \text{bid}_q \)

if \( sales > targetSales \) then

\[ targetROI = targetROI(d) \times \text{INC}_\text{ROI} \]

else

\[ targetROI = targetROI(d) / \text{INC}_\text{ROI} \]

for all \( q \) do

\[ CPC_q = (USP_q - targetROI) \times PrConv_q \]

\[ bid_q = \text{cpc2bid}(CPC_q) \]

**ROI = “profit per sale”**

\[
ROI_q(s_q) = \frac{rev_q(s_q) - cost_q(s_q)}{sales_q(s_q)}
\]

\[
= USP_q - \frac{CPC_q(s_q)\text{numClicks}_q(s_q)}{sales_q(s_q)}
\]

\[
= USP_q - CPC_q(s_q) \frac{1}{PrConv_q}
\]

Borgs, et al. 2007
Optimization Challenge

**GOAL:** to compare the performance of optimization routines, without running too many expensive game experiments

- Run a few real games and log all agents’ decisions and game outcomes
- Pick a candidate optimization routine to substitute for the one used by one of the original agents
  - Send the candidate all the information the original agent received
  - Re-simulate game with candidate’s decisions and all other agents’ decisions
- Repeat using all game logs and all agents, for all candidate optimization routines
Optimization Challenge

<table>
<thead>
<tr>
<th>Agent</th>
<th>Profit</th>
<th>Avg Pos</th>
<th>CPC</th>
<th>ClickPr</th>
<th>ConvPr</th>
<th>Sales in CS</th>
<th>Sales in MS</th>
<th>OverCap</th>
</tr>
</thead>
<tbody>
<tr>
<td>DynamicMCKP</td>
<td>75.93</td>
<td>2.49</td>
<td>0.46</td>
<td>0.21</td>
<td>0.17</td>
<td>0.41</td>
<td>0.85</td>
<td>1.38</td>
</tr>
<tr>
<td>HybridMCKP</td>
<td>75.85</td>
<td>2.42</td>
<td>0.48</td>
<td>0.22</td>
<td>0.17</td>
<td>0.40</td>
<td>0.86</td>
<td>1.40</td>
</tr>
<tr>
<td>ExhaustiveMCKP</td>
<td>75.01</td>
<td>2.70</td>
<td>0.42</td>
<td>0.20</td>
<td>0.14</td>
<td>0.41</td>
<td>0.78</td>
<td>1.45</td>
</tr>
<tr>
<td>EquateROI</td>
<td>73.47</td>
<td>2.33</td>
<td>0.48</td>
<td>0.20</td>
<td>0.14</td>
<td>0.38</td>
<td>0.86</td>
<td>1.40</td>
</tr>
</tbody>
</table>

Optimization routines that equate ROI are competitive with those that equate marginal ROI
## TAC AA Game Experiments

<table>
<thead>
<tr>
<th>Agent</th>
<th>Set 1</th>
<th>Set 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>TacTex</td>
<td>80.76</td>
<td>79.86</td>
</tr>
<tr>
<td>HybridMCKP</td>
<td>77.83</td>
<td></td>
</tr>
<tr>
<td>EquateROI</td>
<td></td>
<td>75.67</td>
</tr>
<tr>
<td>astonTAC</td>
<td>76.30</td>
<td>75.32</td>
</tr>
<tr>
<td>munsey</td>
<td>73.41</td>
<td>72.40</td>
</tr>
<tr>
<td>epflagent</td>
<td>72.43</td>
<td>73.05</td>
</tr>
<tr>
<td>QuakTAC</td>
<td>70.61</td>
<td>70.53</td>
</tr>
<tr>
<td>MetroClick</td>
<td>70.15</td>
<td>68.73</td>
</tr>
<tr>
<td>mertacor</td>
<td>68.31</td>
<td>68.08</td>
</tr>
</tbody>
</table>

Scores in thousands

Optimization algorithms that equate marginal ROI are competitive with those that equate ROI.

Opponent scores and rankings are similar, regardless of our agent.
Model Accuracy

- As model accuracy degrades:
  - performance of model-light algorithms is unchanged
  - model-light algorithms eventually perform better than model-heavy ones
- ClickPr accuracy appears to be more important than CPC
JACK
JAVA Auction Configuration Kit
What is **JACK**?

### Auction Types

- **Homogenous Additive**
  
### Auction Structure

### Value Structure

\[ V(g, \alpha, \beta) = \alpha_1 g_1^\beta_1 + \alpha_2 g_2^\beta_2 + \ldots + \alpha_n g_n^\beta_n \]

**Contract**
- Good1: (3, 5)
- Good2: (10, 20)
- Good3: (6, 15)

**Value**
- [45, 33, 28, 17, 10]

**Time**
- 1, 2, 3, 4, 5
Running an Auction
Running an Auction

Configuration
Files
Running an Auction

Configuration Files

JACK
Running an Auction

- Configuration Files
- JACK

Tools:
- Java
- Python
Running an Auction

Configuration Files

JACK

Java

Python
Running an Auction

Configuration Files
Running an Auction

Configuration
Files

JACK

Java
Python
Running an Auction

Configuration Files

Java

Python

JACK
Running an Auction

Configuration Files

Java

python

JACK

Tuesday, July 24, 12
Running an Auction

Configuration
Files

Java

JACK

Python
Running an Auction

Configuration Files

Log Results

JACK

Java

Python
Auctions

- Simple Auctions
- 1st Price
- 2nd Price
- Ascending
- Descending
Auctions

- Simple Auctions
  - 1st Price
  - 2nd Price
  - Ascending
  - Descending

- Compound Auctions
  - Sequential
  - Simultaneous

![Auction Diagram]

- Simultaneous
  - 2nd
  - Ascen.
  - Descen.

- Sequential
  - 1st
  - Descen.
  - 2nd
Valuation Structure

- **Additive**

  \[ \alpha \in [\alpha_L, \alpha_H] \quad \beta \in [\beta_L, \beta_H] \]

  \[ V(g, \alpha, \beta) = \alpha_1 g_1^{\beta_1} + \alpha_2 g_2^{\beta_2} + \ldots + \alpha_n g_n^{\beta_n} \]

- **Scheduling**

  - # of Goods
  - minValue, maxValue
  - minNeeded, maxNeeded

- **Contracts**

  - Good: min#, max#, minValue, maxValue
  - Good: (#, value)
Dutch Flower Auctions

• We used JACK to model a simplification of the DFA
• Simultaneous and independently sequential descending auctions
• The next TAC Game?
Summary

• Two-phased agent architecture lends itself to two complementary testing frameworks
  • Prediction challenge
  • Optimization challenge

• But it’s not quite so simple ...
  • Decisions impact model accuracy
  • Model accuracy impacts optimization quality
Outline

• Agents for “simple” interdependent markets
  • Simultaneous & Sequential

• Agents for more complex interdependent markets
  • Simulated markets: TAC games
  • Real-world market: Quibids

• Mike: Empirical game-theoretic analysis (EGTA), to predict equilibrium outcomes in these markets
Outline

• Agents for “simple” interdependent markets
  • Simultaneous & Sequential
• Agents for more complex interdependent markets
  • Simulated markets: TAC games
• Real-world market: Quibids

• Mike: Empirical game-theoretic analysis (EGTA), to predict equilibrium outcomes in these markets
Outline

- Agents for “simple” interdependent markets
  - Simultaneous & Sequential
- Agents for more complex interdependent markets
  - Simulated markets: TAC games
  - Real-world market: Quibids
- Mike: Empirical game-theoretic analysis (EGTA), to predict equilibrium outcomes in these markets