Expressive Banner Ad Auctions and Model-Based Online Optimization for Clearing

Craig Boutilier (U. Toronto)
David C. Parkes (Harvard U.)
Tuomas Sanholm (Carnegie Mellon U.)
William E. Walsh (CombineNet, inc.)
Context of the Paper

• Allocation of media channel capacity to ads in order to maximize auctioneer revenue (or advertiser satisfaction)

• Novelty of the paper:
  – Campaign-level expressiveness – branding, impression advertising
  – Stochastic optimization with continuous action space
    • Regret algorithm adapted for continuous action space
Allocation Problem – The Locations

Ad Location
Page of NY Times that X views on Nov 17, 2008 at 17:00

Ad Location
Page of NY Times that Y views on Nov 17, 2008 at 18:00

Ad Location
Page of CNN
Allocation Problem - Location Properties

• One physical location has
  – Static properties
    • the page identity - NY Times main page
    • page category - major news site
    • expected demographics
    • identity and size of the location - top banner, wide skyscraper, micro bar
  – Transient location (or allocation) features:
    • time of day
    • page content
    • the presence of a competitor's ad on the same page

• Each location has associated an attribute vector with these properties.
Allocation Problem – The Contracts

Contract 1 (Flat/Simple):
- $0.50 per CT on NY Times.

Contract 2 (Expressive – e.g., Bonus):
- nothing for CNN impressions if fewer than 10K CTs
- $10,000 if at least 100K CTs are achieved on CNN in 10 days
- $0.50 per CT on NY Times after 1K CTs have been achieved.
- contract valid only for 14 days
- $0.20 per impression up to 30K, $0.50 per impression for more
Allocation Problem - Comparison between Existing Contracts and Expressive Contracts

Existing Contracts - typical level of expressiveness in existing auctions:
- Acceptable attributes
- Per-unit bidding (per-impression/per-clickthrough)
- Budgets
- Single-period expressiveness (e.g., 1 day)

Expressive Contracts - Advertising campaigns express preferences over a sequence of allocations:
- Minimum targets: pay only if 100K impressions in a week
- Tiered preferences: $0.20 per impression up to 30K, $0.50 per impression for more
- Temporal sequencing: at least 20K impressions per day for 14 days
- Substitution: either NYT ($0.90) or CNN ($0.50) but not both
- Smoothness: impressions vary by no more than 20% daily
- Long-term budget: spend no more that $250K in a month

Additional forms of expressiveness:
- Advertiser choice of impression/CT/conversion pricing
- Target audience (e.g., demographics) rather than indirectly via web site properties

Much of the research assumes this level of expressiveness

Greater expressiveness increases revenue/efficiency
The Allocation Problem

• Find optimal allocation of capacity to contracts

Supply: web page requests / channels realization (instantiation)

Demand: contracts

Maximize Revenue
(or Allocative Efficiency / Customer Satisfaction)
The Value of Optimization with Sequential Expressiveness

- Bidder 1: bids $1 on A, $0.50 on B, budget $50K
- Bidder 2: bids $0.50 on A, but not on B, budget $20K
- Traditional first-price auction: $52.3K revenue
The Value of Optimization with Sequential Expressiveness

- **Bidder 1:** bids $1 on A, $0.50 on B, budget $50K
- **Bidder 2:** bids $0.50 on A, budget $20K
- Optimal allocation: $70K revenue

Caveat:
- no equilibrium
- example assumes bids do not change
- see next week’s paper - use VCG.

Supply of A
- 50 KCTs
  - \( t_0 \)

Supply of B
- 10 KCTs
  - \( t_1 \)
  - 10 KCTs
  - \( t_2 \)

Bidder 1: 80 KCTs

Bidder 2: 40 KCTs

Bidder 1: 10 KCTs

- Bidder 1: bids $1 on A, $0.50 on B, budget $50K
Channels

• Channel = aggregation of locations/web pages (e.g., collection of web pages)
• Example:
  – Bid 1: NY Times (NYT), Bid 2: Medical article (Med)
  – Channels: (NYT & Med), (NYT & ¬Med), (¬ NYT & Med)
  – Non-NYT pages grouped together, non-Med pages grouped together
• Lossless aggregation of several locations in a channel - classify intelligently (open problem)
• Avoid “Curse of dimensionality” (R. Bellman)/“exponential explosion”
  – use lossy abstraction technique, i.e., aggregate even more the channels --> suboptimality.
  • Example: aggregate channel classes \{(NYT & Med), (NYT & ¬Med)\} in \{NYT\}.
David’s Remarks During the Presentation

• Regarding the validity check in the dispatching - done such it allocates the channel only to valid contracts (after doing lossy abstraction)
  – Can increase the revenue, but no theory how much better
  – architecture - optimize & dispatch
  – no point in giving (NY & !Med) to somebody who wants (Med) - you improve the quality, but does not become optimum policy
    • Example: dispatcher parameterization – when supply has uniform distribution: 20% Med, 80% non-Med
    • Assume bidder with preference only for Med
      – Then dispatcher should give 5 times more to the winner of the optimization to be able to meet the “minimum target” contract

• Channel aggregation
  – clustering - machine learning technique
  – value based - more exactly:
    • optimization performed at a course abstraction
    • multiply/improve the quality the optimization by looking to split a node in order to improve the optimality
      – think of it like optimization on a decision tree
      – optimization in one big node
        » split in 2: NY, !NY - greedy myopic to split and then prove about optimality
      – wrapper around
      – bidder should not be constrained on how they bid
        » do the abstraction given the bid.
Contract Model

• Model as state space system - Markov Decision Process (MDP)
  • Markovian property: the evolution of the system depends only on current state
  • state = “sufficient statistic summarizing relevant aspects”
  • reward in the MDP = revenue in associated period

• Most refined model - represent contract through its history - log with impression attributes per period.
• If contracts richer (more info/constraints per contract) then MDP model becomes more complex
Example of Contract MDP Model

• Suppose: only 1 channel; bid 1K$/1K impressions

Notes:
- 1 state transition = 1 period
- numbers in nodes = number of impressions (in K) generated in the corresponding time period
- \( x^t_{ij} \) = fraction of allocation of contract to the channel
- MDP transitions for period 2 derived from following \( P^s \): 3K – 0.2, 6K – 0.4, 9K - 0.4
Allocation Problem – Mathematical Formulation

• $C = \text{set of channels}$

• $B = \text{set of bids/contracts}$

• $T = \text{maximum horizon (of contracts)}$

• Decision variables: $\{x_{ij}^t : i \leq C, j \leq B, t \leq T \}$
  
  – $x_{ij}^t = \text{percentage of channel } i \text{ assigned to contract } j \text{ at period } t$.

  – Assuming $C^{\text{avg}} \text{ relevant channels per contract}$, we have $O(BC^{\text{avg}})$ decision variables per period.
Allocation Problem - Mathematical Formulation

• “Optimize-and-dispatch” framework

- Supply model: Page hit (channel size) probability distribution ($P^S$)
- Demand model: Contract description and probability distribution ($P^D$) over location-price pairs

Auctioneer: Revenue Maximization

$\pi = <\pi^1, ..., \pi^T> (\pi^t:S \to X)$ s.t.

maximize: $E \left[ \sum_{j \in B} R(j, A_{j,1:T}) \mid \pi \right]$

where $A_{j,1:T} = \text{random var with set of locations assigned to contract } j$
MDP model

• State space:
  – parallel composition of each contract MDP model
    • \( S = \prod_{j \in B} S_j \) where \( S_j = \text{contract MDP model} \)

• An action of the MDP is a valuation of the decision variables \( x_{ij}^t \):
  – \( X = \{ x : B \times C \rightarrow [0, 1] \mid \sum_{j \in B} x_{ij} \leq 1, \forall i \in C \} \)

• Transition probabilities:
  – \( P_t(s' \mid x, s) \)
    • time dependent
    • can be defined using \( P^S \) (supply distribution), CT rates, \( P^D \) (demand distribution)
Standard Equations for Optimization

• Bellman’s equations:
  – value function:

  \[ V^{T+1}(s) = R(s) \] and for \( t \in [1, T] \) we have

  \[ V^t(s) = \max_x R^t(s, x) + \sum_{s' \in S} P^t(s'|x, s)V^{t+1}(s') \]

  – optimal policy:

  \[ \pi^t(s) = \arg\max_x R^t(s, x) \sum_{s' \in S} P^t(s'|x, s)V^{t+1}(s') \]

  • Computed through backward induction
Allocation Problem Optimization

• Infeasible to do offline (MDP) optimization
  – Huge state space – cross product of individual campaign states
    • order of 10K channels, 1M bids → policy $\pi$ would have in the order of 1 trillion entries, for the time horizon
  – High-dimensional continuous action space

• Two Online Optimization Methods Proposed
  – Expectation-based (re)optimization
  – Online stochastic optimization – Regrets algorithm

• Have to optimize online at least because of the evolving state of the contract --> reoptimize.
Expectation-based (re)optimization #1

- $z_i^t = E[\text{size of channel } i \text{ at time } t]$
- MILP
  - $\max$: reward (bids, constants $z_i$, decision vars $x_{ij}$)
  - $x_{ij}^t \geq 0$, for all periods $t$
  - $\sum_j x_{ij}^t \leq 1$, for all channels $i$, periods $t$
  - Integer vars are, for example, variables that indicate threshold number of CTs have been achieved.
- MILP solvers customized for auctions can solve very large problems ([Sandholm_2007])
  - order of 10K channels, 1M bids, 100K side constraints
- For trivial contracts, can use also min-cost (cost = - reward) max flow problem on bipartite graph with bids and channels.
- Note: this expectation-based (re)optimization is similar to the one in [Lahaie_AAAI2008]
Expectation-based (re)optimization #2

• Solutions may be far from optimal if supply distributions have high variance
  – Does not adequately account for risk
  – Can be mitigated by periodic reoptimization
Online stochastic optimization

- *Online* – compute using Continuous REGRETS only the next action, rather than entire policy
  - Informed by what we might do in the future
  - Recompute at each time period

- Extremely effective when good deterministic algorithms exist

- Requires that domain uncertainty is exogenous
  - Distribution of future events do not depend on decisions
    - Allows to explore all the way into the future, and then decide what to do.
    - If not exogenous - the uncertainty (of the supply of impressions and new demands) of the future evolution depends on the actions - there is a class of very fast approx algorithms one can use.
REGRETS Algorithm

• Proposed in [Bent_AAAI2004]
• Sample based technique (see [Bishop_2006], chapters 8.1.2, 11)
  – Monte Carlo algorithm

• Basically:
  – Sample – Predict what happens in the future - based on the data from the past
    • Use Poisson distribution model or data driven
  – Combine the solutions based on future realizations
    • Continuous actions: It has embedded all the optimizations in one
REGRETS Algorithm - Sampling

Scenario/Trace/Realization of channel sizes ($z_i$) over time

Sample $K$ scenarios:

- $t$: Compute with Regrets optimum decision at period $t$
- $t+1$: Compute with Regrets optimum decision at period $t+1$
- $T$: Periods
REGRETS Algorithm

Input: Current time $t$, decisions in current time $\mathcal{X}$, $K$ scenarios, $s^t$ current state.

for $x \in \mathcal{X}$, $f(x) \leftarrow 0$
for $k = 1 \ldots K$:
    $\omega_k \leftarrow \text{GetSample}(t + 1, T)$
    $x^* \leftarrow \text{OptimalSoln}(s_t, \omega_k)$
    $f(x^*(t)) \leftarrow f(x^*(t)) + w(x^*)$
    for $x \in \mathcal{X} \setminus \{x^*(t)\}$
        $f(x) \leftarrow f(x) + w(x^*) - \text{Regret}(x^*, x, s_t, \omega_k)$

Output: Decision for time $t$ is $\arg \max \{f(x) : x \in \mathcal{X} \}$
along with estimated expected value $f(x)/K$.

Where:

$$\text{Regret}(x^*, x, s_t, \omega_k) \geq \overline{V}^t_{\pi}(s_t) - \overline{Q}^t_x(s_t)$$

- $f = \text{evaluation function [Bent\_AAAI2004]}$
- $f(x)/K = \text{estimate of the Q-value of x, in crt state}$
REGRETS Algorithm
([Bent_AAAI2004])

```
Time t
Action       Value

x_{t,1}         f(x_{t,1})
```

Choose \( x_{t,i} \) that maximizes \( f(x_{t,i}) \)

```
x_{t,2}         f(x_{t,2})

\ldots

x_{t,3}         f(x_{t,3})

\ldots

x_{t,n}         f(x_{t,n})
```

```
\langle \hat{x}_{1}^{f}, \hat{x}_{1}^{t+1}, \ldots, \hat{x}_{1}^{T} \rangle

\langle \hat{x}_{2}^{f}, \hat{x}_{2}^{t+1}, \ldots, \hat{x}_{2}^{T} \rangle

\ldots

\langle \hat{x}_{K}^{f}, \hat{x}_{K}^{t+1}, \ldots, \hat{x}_{K}^{T} \rangle
```

Sample \( z_1 \)
Optimal Soln.

```
\langle x_{1}^{t}, x_{1}^{t+1}, \ldots, x_{1}^{T} \rangle

\langle x_{2}^{t}, x_{2}^{t+1}, \ldots, x_{2}^{T} \rangle

\ldots

\langle x_{K}^{t}, x_{K}^{t+1}, \ldots, x_{K}^{T} \rangle
```

Sample \( z_2 \)
Optimal Soln.

Sample \( z_K \)
Optimal Soln.
REGRETS Algorithm

([Bent_AAAI2004])

Lower bound on Q-values for action $x^t$ at time $t$

$$f(x^t) = \sum \left\{ \begin{array}{l}
\langle x^t, \hat{x}^{t+1}_1, \ldots, \hat{x}^T_1 \rangle \\
Q_1^t(x') \\
\langle x^t, \hat{x}^{t+1}_2, \ldots, \hat{x}^T_2 \rangle \\
Q_2^t(x') \\
\vdots \\
\langle x^t, \hat{x}^{t+1}_K, \ldots, \hat{x}^T_K \rangle \\
Q_K^t(x')
\end{array} \right\}$$

$z_1^t z_1^{t+1} z_1^{t+2} z_1^{t+3} z_1^{t+4} \ldots z_1^T$

Sample $z_1$
Optimal Soln.

$\langle \hat{x}_1^t, \hat{x}_1^{t+1}, \ldots, \hat{x}_1^T \rangle$

$z_2^t z_2^{t+1} z_2^{t+2} z_2^{t+3} z_2^{t+4} \ldots z_2^T$

Sample $z_2$
Optimal Soln.

$\langle \hat{x}_2^t, \hat{x}_2^{t+1}, \ldots, \hat{x}_2^T \rangle$

$z_K^t z_K^{t+1} z_K^{t+2} z_K^{t+3} z_K^{t+4} \ldots z_K^T$

Sample $z_K$
Optimal Soln.

$\langle \hat{x}_K^t, \hat{x}_K^{t+1}, \ldots, \hat{x}_K^T \rangle$
Extending REGRETS for Continuous Action Space

• REGRETS cannot be directly applied to ad auctions
  – Requires set of possible first-period decisions to be small and discrete
  – Our dispatch policies are *continuous*
  – Even a discretization of our continuous decision space would be huge: dimensionality = $|C| |B| |\text{Discretization}|$

• We leverage deterministic MILP algorithm to (approximately) solve stochastic optimization problem
  – $K + 1$ MILPs
  – 1 MILP per each of $K$ samples
  – 1 MILP for aggregation
Continuous REGRETS

Combining MILP

Instead of enumerating over $X$ to find optimal allocation (like in standard REGRETS), we solve this MILP:

$$\max \frac{1}{K} \sum_{k \leq K} Q^t_k(x^t)$$

$Q^t_k(x^t) = $ lower bound of $Q$-value (because suboptimal)

Sample $z_1$
Optimal Soln. from MILP:
$$\langle x_1^t, \hat{x}_1^{t+1}, \ldots, \hat{x}_1^T \rangle$$

Sample $z_2$
Optimal Soln. from MILP:
$$\langle \hat{x}_2^t, \hat{x}_2^{t+1}, \ldots, \hat{x}_2^T \rangle$$

Sample $z_K$
Optimal Soln. from MILP:
$$\langle \hat{x}_K^t, \hat{x}_K^{t+1}, \ldots, \hat{x}_K^T \rangle$$
Handling Budgets (and Other Constraints)

• In combining MILP, it is too constraining to impose budget constraint for contract $j$ in each scenario:

$$Q^t_{k,j}(x^t) \leq D_j$$

• Allocation may exceed budget in some scenarios
  – But this is ok, since we will be revising policy as time goes on, and will stop dispatching when reach budget

• Instead, don’t count payments that exceed the budget (dispatcher responsibility):

$$Q^t_{k,j}(x^t) = \max\left(\hat{Q}^t_{k,j}(x^t), D_j\right)$$
Empirical Evaluation

• Two preference types:
  – Flat – per-unit bids, budget
  – Bonus – per-unit bids plus bonus, budget

• Two supply distributions:
  – Unimodal – Poisson
  – Bimodal – two Poissons with persistence

• 10 random instances per combination of preference/supply
  – 10 channels, 50 bidders
  – 100 supply realizations of each instance

• Four methods:
  – Bid-all (flat only)- Traditional auctions (1st price), bid on all channels of interest
  – Myopic (flat only)- Traditional auctions (1st price), myopically maximize ROI
  – Deterministic - Expressive auctions, deterministic optimization
  – Stochastic - Expressive auctions, stochastic optimization, 10 samples

• Measured revenue
## Revenue: Flat Bids

<table>
<thead>
<tr>
<th>Method</th>
<th>Unimodal supply</th>
<th>Bimodal supply</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bid-all</td>
<td>25,687 ± 436</td>
<td>14,004 ± 141</td>
</tr>
<tr>
<td>Myopic</td>
<td>30,256 ± 437</td>
<td>15,890 ± 175</td>
</tr>
<tr>
<td>Deterministic</td>
<td>42,365 ± 581</td>
<td>22,385 ± 227</td>
</tr>
<tr>
<td>Stochastic</td>
<td>42,237 ± 581</td>
<td>22,774 ± 238</td>
</tr>
</tbody>
</table>

Conclusions: Better to use expressive auction mechanism than simple one. No clear advantage of Stochastic optimization against Expectation based optimization.
## Revenue: Bonus Bids

<table>
<thead>
<tr>
<th>Method</th>
<th>Unimodal supply</th>
<th>Bimodal supply</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deterministic</td>
<td>100,266 ± 3,555</td>
<td>55,901 ± 1,887</td>
</tr>
<tr>
<td>Stochastic</td>
<td>149,423 ± 3,204</td>
<td>65,065 ± 2,356</td>
</tr>
</tbody>
</table>

Conclusions: Now Stochastic optimization performs better than Expectation based optimization.

“Furthermore, to the extent that increased revenue reflects improved allocative efficiency, then this advantage would also be expected to extend to efficiency.”
Critiques

• There is no Incentive compatibility – because use FP auction
  – For non-expressive mechanisms – ROI based approach
    - remember [Borgs_WWW2007]

• Experiments with supply/demand models not backed up by real data
  – Next to impossible to get real data, since companies (e.g., Yahoo) do manually long-term contracts.

• Few details on MDP modeling.
Food for Thought

• Find a good (i.e. competitive) classification method for the contracts. Think of:
  – Machine learning clustering techniques.

• Maybe possible to use MDP optimization, even if state space is infinite.
Thank you! Questions?
References