# Online Algorithms

## The BALANCE Algorithm

Mining of Massive Datasets
Leskovec, Rajaraman, and Ullman
Stanford University

# Adwords Problem

- **Given:**
  - A set of bids by advertisers for search queries
  - A click-through rate for each advertiser-query pair
  - A budget for each advertiser (say for 1 day, month...)
  - A limit on the number of ads to be displayed with each search query
- **Respond to each search query with a set of advertisers such that:**
  - The size of the set is no larger than the limit on the number of ads per query
  - Each advertiser has bid on the search query
  - Each advertiser has enough budget left to pay for the ad if it is clicked upon

# Dealing with Limited Budgets

- **Our setting: Simplified environment**
  - There is **1** ad shown for each query
  - All advertisers have the same budget ***B***
  - All ads are equally likely to be clicked
  - Value of each ad is the same (=**1**)
- **Simplest algorithm is greedy:**
  - For a query pick any advertiser who has bid **1** for that query
  - **Competitive ratio of greedy is 1/2**

# Bad Scenario for Greedy

- **Two advertisers A and B**
  - ***A*** bids on query ***x***, ***B*** bids on ***x*** and ***y***
  - Both have budgets of **$4**
- **Query stream: *x x x x y y y y***
  - Worst case greedy choice: ***B B B B*** **_ _ _ _**
  - Optimal:  **A A A A B B B B**
  - **Competitive ratio = ½**
- **This is the worst case!**
  - **Note:** Greedy algorithm is deterministic – it always resolves draws in the same way

# BALANCE Algorithm [MSVV]

- **BALANCE** Algorithm by Mehta, Saberi, Vazirani, and Vazirani
  - **For each query, pick the advertiser with the largest unspent budget**
  - Break ties arbitrarily (**but in a deterministic way**)

# Example: BALANCE

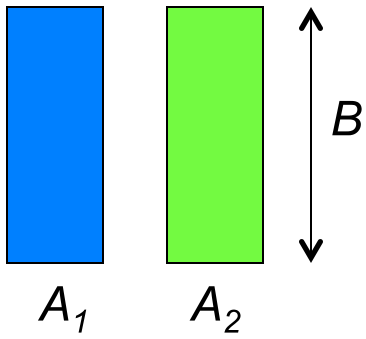
- Two advertisers A and B
  - **A** bids on query ***x***, **B** bids on ***x*** and ***y***
  - Both have budgets of **$4**
- Query stream: ***x x x x y y y y***
- BALANCE choice: **A B A B B B _ _**
  - Optimal: **A A A A B B B B**
- Competitive ratio = ¾
  - **For BALANCE with 2 advertisers**

# Analyzing 2-advertiser BALANCE

- **Consider simple case**
  - **2** advertisers, $\mathbf{A_1}$ and $\mathbf{A_2}$, each with budget **B** ($\geq$1)
  - Optimal solution exhausts both advertisers' budgets

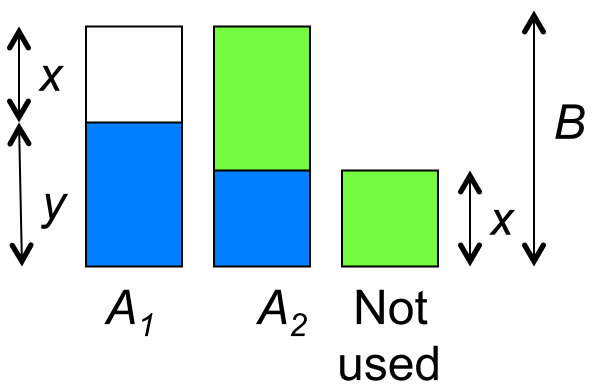
- **BALANCE must exhaust at least one advertiser's budget:**
  - **If not, we can allocate more queries**
  - Assume BALANCE exhausts $A_2$'s budget

# Analyzing Balance




Queries allocated to $\boldsymbol{A_1}$ in the optimal solution

Queries allocated to $\boldsymbol{A_2}$ in the optimal solution

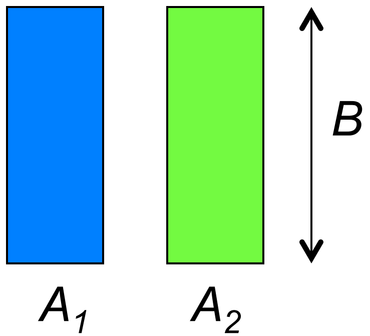



Optimal revenue = **2B**
Balance revenue **= B+y**
Goal: Show that $y \geq B/2$

**Case 1**: BALANCE assigns at least B/2 blue queries to $A_1$. So $y \geq B/2$.

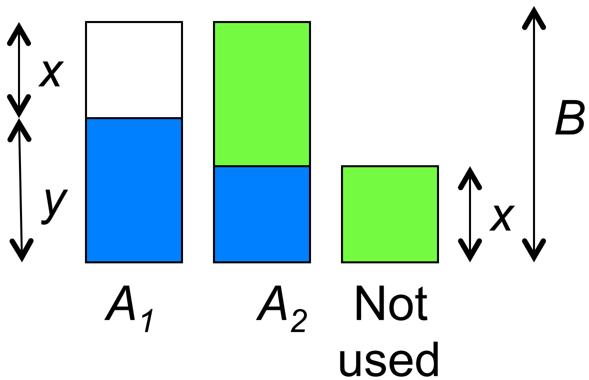
**Case 2**: BALANCE assigns more than B/2 blue queries to $A_2$.
Consider the last blue query assigned to $A_2$.
At that time, $A_2$'s unspent budget must have been at least as big as $A_1$'s.
That means at least as many queries have been assigned to $A_1$ as to $A_2$.
At this point, we have already assigned at least B/2 queries to $A_2$.
So $y \geq B/2$.

# Analyzing BALANCE




Queries allocated to $A_1$ in the optimal solution

Queries allocated to $A_2$ in the optimal solution




Optimal revenue OPT = **2B**
Balance revenue BAL **= B+y**

We have shown that $y \geq B/2$
$BAL \geq B+B/2 = 3B/2$
$BAL/OPT \geq 3/4$

# BALANCE: General Result

- **In the general case, worst competitive ratio of BALANCE is 1–1/e = approx. 0.63**
  - Interestingly, no online algorithm has a better competitive ratio!
- **Let's see the worst case example that gives this ratio**