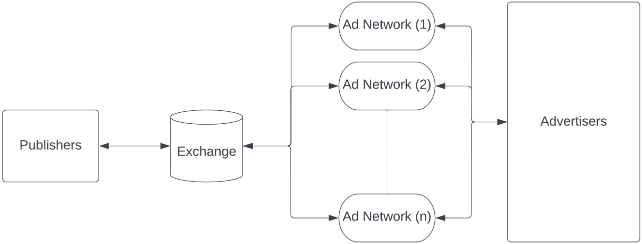
AdX Agent Design

CSCI 1440/2440 Algorithmic Game Theory

October 29, 2025

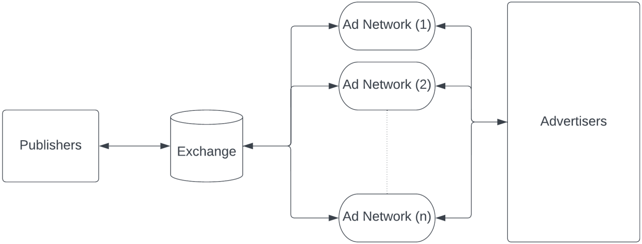
Motivation: Ad Exchanges

- In contrast to sponsored search web users also see display ads
- Advertisers do not want to have to negotiate with publishers



Motivation: Ad Exchanges

- Ad exchange exist to connect publishers and advertisers
- Publishers lists there advertising impressions on the exchange
- Ad networks bid on behalf of advertisers for impressions



DoubleClick

- Acquired by Google in 2007 for $3.1 Billion
- Google used DoubleClick to create there own advertising exchange which later became AdSense
- In 2022, Google AdSense generated $32.8 Billion in revenues

AdX Game

- A market with multiple copies of heterogeneous goods: **users' impression opportunities**
- Your agent's objective: procure **enough targeted** impressions at the **lowest** possible cost

Two sources of uncertainty

- Users (Supply)
- Competition (Demand)

Lab Variants of the AdX Game

- One-Day, One-Campaign
- Two-Days, Two-Campaigns

Modeling the One-Day Game.

Game Elements:

Modeling the One-Day Game.

Game Elements:

- A set of **market segments** M of size m

Modeling the One-Day Game.

Game Elements:

- A set of **market segments** M of size m
- a probability mass function $\pi = \langle \pi_1, \pi_2, \dots, \pi_m \rangle$ s.t. π_j is the probability of drawing a **user** from the jth market segment (e.g., old, female)

Modeling the One-Day Game.

Game Elements:

- A set of **market segments** M of size m
- a probability mass function $\pi = \langle \pi_1, \pi_2, \ldots, \pi_m \rangle$ s.t. π_j is the probability of drawing a **user** from the jth market segment (e.g., old, female)
- A **campaign** $c = \langle r, m, b \rangle$

  demands $r \sim R(\cdot)$ impressions (i.e., reach)
  from a market segment $m \in M$ such that $m \sim G(\cdot)$
  for which it will earn budget $b \sim B(\cdot)$

Modeling the One-Day Game (cont).

A **One-Day Game**, has N agents, each with a **single** campaign: $c_i = \langle r_i, m_i, b_i \rangle$, for all $i \in \{1, \ldots, N\}$

Let $\boldsymbol{x} = \langle x_1, x_2, \ldots, x_m \rangle$ be a bundle of impressions.
The **utility** u_i of agent i, as a function of bundle $\boldsymbol{x}$, is given by:

$$u_i(\boldsymbol{x}, c_i) = \rho(\mu(\boldsymbol{x}, c_i)) - p(\boldsymbol{x})$$

Modeling the One-Day Game (cont).

A **One-Day Game**, has N agents, each with a **single** campaign: $c_i = \langle r_i, m_i, b_i \rangle$, for all $i \in \{1, \ldots, N\}$

Let $\boldsymbol{x} = \langle x_1, x_2, \ldots, x_m \rangle$ be a bundle of impressions.
The **utility** u_i of agent i, as a function of bundle $\boldsymbol{x}$, is given by:

$$u_i(\boldsymbol{x}, c_i) = \rho(\mu(\boldsymbol{x}, c_i)) - p(\boldsymbol{x})$$

$p(\boldsymbol{x})$ is the total **cost** of bundle $\boldsymbol{x}$

Modeling the One-Day Game (cont).

A **One-Day Game**, has N agents, each with a **single** campaign: $c_i = \langle r_i, m_i, b_i \rangle$, for all $i \in \{1, \ldots, N\}$

Let $\boldsymbol{x} = \langle x_1, x_2, \ldots, x_m \rangle$ be a bundle of impressions.
The **utility** u_i of agent i, as a function of bundle $\boldsymbol{x}$, is given by:

$$u_i(\boldsymbol{x}, c_i) = \rho(\mu(\boldsymbol{x}, c_i)) - p(\boldsymbol{x})$$

$p(\boldsymbol{x})$ is the total **cost** of bundle $\boldsymbol{x}$

$$\mu(\boldsymbol{x}, c_i) = \sum_{m \in M} x_m \mathbb{1}_{m = m_i}$$

Modeling the One-Day Game (cont).

A **One-Day Game**, has N agents, each with a **single** campaign: $c_i = \langle r_i, m_i, b_i \rangle$, for all $i \in \{1, \ldots, N\}$

Let $\boldsymbol{x} = \langle x_1, x_2, \ldots, x_m \rangle$ be a bundle of impressions.
The **utility** u_i of agent i, as a function of bundle $\boldsymbol{x}$, is given by:

$$u_i(\boldsymbol{x}, c_i) = \rho(\mu(\boldsymbol{x}, c_i)) - p(\boldsymbol{x})$$

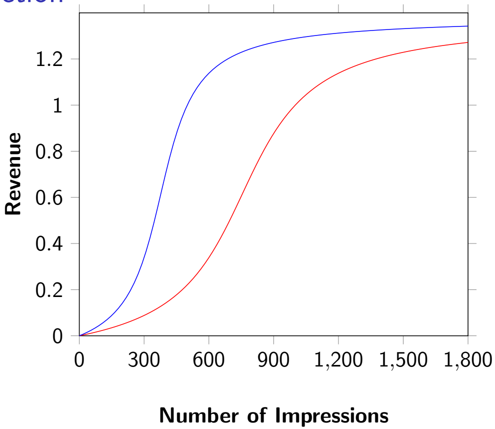
$p(\boldsymbol{x})$ is the total **cost** of bundle $\boldsymbol{x}$

$$\mu(\boldsymbol{x}, c_i) = \sum_{m \in M} x_m \mathbb{1}_{m = m_i}$$

$\rho(\cdot)$ is a **revenue function** mapping impressions to revenue

Modeling the One-Day Game (cont).

Revenue function



Sample Revenue Functions when Reach equals 500 (blue) and 1000 (red).

Modeling the One-Day Game (cont).

Game Dynamics

Stage 1: Agent i learns its own type c_i, but not others' types.

Stage 2: All agents compute and submit their bids.

Stage 3: The K users arrive in a random order

$$\langle \boldsymbol{m} \rangle = \left\langle m^1, m^2, \ldots, m^K \right\rangle, \text{ where } m^k \sim \pi$$

For each user k that arrives, a second-price auction is held.

The game ends and utilities are realized.

Modeling the One-Day Game (cont).

Strategies

Agent i's **strategy** $s_i(c_i)$ maps a campaign c_i to a tuple $\langle \boldsymbol{b}_i, \boldsymbol{l}_i \rangle$.

Modeling the One-Day Game (cont).

Strategies

Agent i's **strategy** $s_i(c_i)$ maps a campaign c_i to a tuple $\langle \boldsymbol{b}_i, \boldsymbol{l}_i \rangle$.

- $\boldsymbol{b}_i = \langle b_{i1}, b_{i2}, \cdots, b_{im} \rangle$ is a bid vector, where $b_{ij} \in \mathbb{R}_+$ is agent i's bid for the jth market segment

Modeling the One-Day Game (cont).

Strategies

Agent i's **strategy** $s_i(c_i)$ maps a campaign c_i to a tuple $\langle \boldsymbol{b}_i, \boldsymbol{l}_i \rangle$.

- $\boldsymbol{b}_i = \langle b_{i1}, b_{i2}, \cdots, b_{im} \rangle$ is a bid vector, where $b_{ij} \in \mathbb{R}_+$ is agent i's bid for the jth market segment
- $\boldsymbol{l}_i = \langle l_{i1}, l_{i2}, \cdots, l_{im} \rangle$ is a limit vector, where $l_{ij} \in \mathbb{R}_+$ is the total spending limit of agent i in auctions matching the jth market segment

Modeling the One-Day Game (cont).

Strategies

Agent i's **strategy** $s_i(c_i)$ maps a campaign c_i to a tuple $\langle \boldsymbol{b}_i, \boldsymbol{l}_i \rangle$.

- $\boldsymbol{b}_i = \langle b_{i1}, b_{i2}, \cdots, b_{im} \rangle$ is a bid vector, where $b_{ij} \in \mathbb{R}_+$ is agent i's bid for the jth market segment
- $\boldsymbol{l}_i = \langle l_{i1}, l_{i2}, \cdots, l_{im} \rangle$ is a limit vector, where $l_{ij} \in \mathbb{R}_+$ is the total spending limit of agent i in auctions matching the jth market segment

Notation

Denote by $\boldsymbol{s}_{-i}$ the strategies of agents other than agent i

Modeling the One-Day Game (cont).

The bundle $\boldsymbol{x} = \boldsymbol{x}(s_i, \boldsymbol{s}_{-i}, \boldsymbol{m})$ procured by agent i depends on:

- its strategy s_i
- other agents' strategies $\boldsymbol{s}_{-i}$
- and the realization of the users (types and their ordering)

Modeling the One-Day Game (cont).

The bundle $\boldsymbol{x} = \boldsymbol{x}(s_i, \boldsymbol{s}_{-i}, \boldsymbol{m})$ procured by agent i depends on:

- its strategy s_i
- other agents' strategies $\boldsymbol{s}_{-i}$
- and the realization of the users (types and their ordering)

We can now state our goal!

Given $\boldsymbol{s}_{-i}$, find s_i^* that maximizes i's interim expected utility:

$$s_i^* \in \arg\max_{s_i} \left\{ \mathop{\mathbb{E}}_{\substack{\boldsymbol{c}_{-i} \\ \boldsymbol{m} \sim \pi^K}} \left[u_i(\boldsymbol{x}(s_i(c_i), \boldsymbol{s}_{-i}(\boldsymbol{c}_{-i}), \boldsymbol{m}); c_i) \right] \right\}$$

Building an interim expected utility maximizing agent

Are we done? (Hint: nope, not even close...)

- Can we evaluate $\boldsymbol{s}_{-i}(\boldsymbol{c}_{-i})$?
- Perhaps, if we know (or learn) $\boldsymbol{s}_{-i}$.
- But now, we don't! So we'll make assumptions...
- Make sure you clearly state any assumptions you make!
- You should also try to justify your assumptions, even when their only justification is computational tractability.

Building an interim expected utility maximizing agent

Are we done? (Hint: nope, not even close...)

- Can we evaluate $\boldsymbol{s}_{-i}(\boldsymbol{c}_{-i})$?
- Perhaps, if we know (or learn) $\boldsymbol{s}_{-i}$.
- But now, we don't! So we'll make assumptions...
- Make sure you clearly state any assumptions you make!
- You should also try to justify your assumptions, even when their only justification is computational tractability.

ASSUMPTION!

We will be making assumptions about the behavior of other agents in the game (as well as other assumptions)!

Supply assumption

SUPPLY ASSUMPTION!

Assume that, for each market segment $m \in M$, there are exactly as many users as expected according to distribution π, i.e., the number of users belonging to market segment m is $K\pi_m$.

We reduce the number of random events to think about by assuming a fixed, deterministic supply.

Demand assumption

Demand Assumption!

Assume (for the moment), we know other agents' campaigns $\boldsymbol{c}_{-i}$.

In reality we know only the distribution of other agents' campaigns. We work with the demand assumption in what follows, and later discuss ways to lift this assumption.

Game of complete information

Supply assumption + Demand assumption =

Game of complete information

Equilibrium Approach

High-Level Idea

- Compute an equilibrium and use it as your agent's prediction of the outcome of the game.
- Program your agent to play its part.
- Works well when an equilibrium is unique!

Equilibrium Approach

High-Level Idea

- Compute an equilibrium and use it as your agent's prediction of the outcome of the game.
- Program your agent to play its part.
- Works well when an equilibrium is unique!

Equilibrium Assumption!

An equilibrium is a good prediction of the outcome of the game.
(Justification: Repeated play would lead to an equilibrium!)

Two Equilibrium Approaches

Market Equilibrium Approach

- Compute a competitive (or Walrasian) equilibrium of the market induced by the game.
- Market equilibria do not always exist. May need to make simplifying assumptions (e.g., divisibility/Fisher markets).

Game-Theoretic Approach

- Predict a game-theoretic equilibrium, such as Bayes-Nash or EPNE (if it exists)
- But computing Nash equilibria is computationally complex.
- Iterative approach (may not converge): predict the behavior of the other agents in the game, and then compute your agent's move as a best-response to this prediction. Repeat.

Game-Theoretic Approach

High-Level Idea

Compute a game-theoretic equilibrium, and then program your agent to **bid its part** of this equilibrium.

Best-reply dynamics

- Predict the behavior of the other agents in the game
- Compute your agent's best-response to this prediction
- Repeat

Game-Theoretic Approach

Deterministic bid assumption!

Suppose agent i, for every segment matching its campaign:

- bids some fraction of budget over reach: i.e., $\beta_i = \frac{b_i}{r_i}$
- bids $\rho_i \beta_i$, where $\rho_i \in [0, 1]$ is a bid shading parameter
- sets its spending limit equal to its campaign's budget b_i

Game-Theoretic Approach

Deterministic bid assumption!

Suppose agent i, for every segment matching its campaign:

- bids some fraction of budget over reach: i.e., $\beta_i = \frac{b_i}{r_i}$
- bids $\rho_i \beta_i$, where $\rho_i \in [0, 1]$ is a bid shading parameter
- sets its spending limit equal to its campaign's budget b_i

Compute best response

- Now we **have** $\boldsymbol{s}_{-i}(\boldsymbol{c}_{-i})$!

Game-Theoretic Approach

Deterministic bid assumption!

Suppose agent i, for every segment matching its campaign:

- bids some fraction of budget over reach: i.e., $\beta_i = \frac{b_i}{r_i}$
- bids $\rho_i \beta_i$, where $\rho_i \in [0, 1]$ is a bid shading parameter
- sets its spending limit equal to its campaign's budget b_i

Compute best response

- Now we **have** $\boldsymbol{s}_{-i}(\boldsymbol{c}_{-i})$!
- So given $\boldsymbol{c}_{-i}$, we can simulate the game.

Game-Theoretic Approach

Deterministic bid assumption!

Suppose agent i, for every segment matching its campaign:

- bids some fraction of budget over reach: i.e., $\beta_i = \frac{b_i}{r_i}$
- bids $\rho_i \beta_i$, where $\rho_i \in [0, 1]$ is a bid shading parameter
- sets its spending limit equal to its campaign's budget b_i

Compute best response

- Now we **have** $\boldsymbol{s}_{-i}(\boldsymbol{c}_{-i})$!
- So given $\boldsymbol{c}_{-i}$, we can simulate the game.
- Can we solve for an optimal best response given $\boldsymbol{c}_{-i}$?

Game-Theoretic Approach (cont.)

Let's analyze a **single market segment** m with initial supply K_m.
Assume we know (e.g., have sampled) $\boldsymbol{c}_{-i}$.

Order other agents' bids: $\rho_1\beta_1 \geq \rho_2\beta_2 \geq \cdots$
The kth bidder gets x_k impressions and pays $p_k = x_k\rho_{k+1}\beta_{k+1}$.

$$x_k = \underbrace{\min\left(\underbrace{\frac{b_k}{\rho_{k+1}\beta_{k+1}}}_{\text{affordable imp'ns}}, \underbrace{K_m - \sum_{t=1}^{k-1} x_t}_{\text{remaining supply}} \right)}_{\substack{k\text{th agent either spends its entire budget} \\ \text{or is allocated all the remaining supply} \\ \text{(or both)}}}$$

Game-Theoretic Approach (cont.)

Algorithm 1 Simulate Auctions

```
procedure GETALLOCATIONANDPAYMENT(a, k, b_a)
    x_i ← 0, p_i ← 0 for all i; currentSupply ← K_m
    Insert a's bid (and hence a) into kth position,
        i.e., b ∈ (ρ_k β_k, ρ_{k−1} β_{k−1})
    for i ∈ [k] do
        x_i ← min { b_i / (ρ_{i+1} β_{i+1}), currentSupply }
        currentSupply ← currentSupply − x_i
    p_k ← x_k ρ_{k+1} β_{k+1}
    return (x_k, p_k)
```

Game-Theoretic Approach (cont.)

Choose optimal position k^*

$$k^* \in \operatorname*{arg\,max}_{k \in \{1,2,\ldots,\}} \{u(x_k, p_k)\} \quad ,$$

where $u(x_k, p_k)$ is your utility for x_k impressions at price p_k.

Game-Theoretic Approach (cont.)

Choose optimal position k^*

$$k^* \in \operatorname*{arg\,max}_{k \in \{1,2,\ldots,\}} \{u(x_k, p_k)\} \quad ,$$

where $u(x_k, p_k)$ is your utility for x_k impressions at price p_k.

Choose your bid: $b \in \left(\rho_{k^*}\left(\frac{b_{k^*}}{r_{k^*}}\right), \rho_{k^*-1}\left(\frac{b_{k^*-1}}{r_{k^*-1}}\right)\right)$

Game-Theoretic Approach (cont.)

$A = \{a_1, a_2, a_3\}$. Market segment m with $K_m = 100$. No reserve.

a_1: $b_1 = 40, r_1 = 10, \rho_1 = 0.75$ and bid $= 3$

a_2: $b_2 = 60, r_2 = 30$ and bid $= 2$

a_3: $b_3 = 20, r_3 = 20$ and bid $= 1$

$\textsc{getAllocationAndPayment}(a, k, b_a = 30)$ Reach $= 20$

$k = 1$ $x_a = \min\{30/3, 100\} = 10;\ p_a = 3;\ u_a = 1.5(10) - 30 = -15$

$k = 2$ $x_1 = \min\{40/2.5, 100\} = 16$

$x_a = \min\{30/2, 84\} = 15;\ p_a = 2;\ u_a = 1.5(15) - 30 = -7.5$

$k = 3$ $x_1 = \min\{40/2, 100\} = 20$

$x_2 = \min\{60/1.5, 80\} = 40$

$x_a = \min\{30/1, 40\} = 30;\ p_a = 1;\ u_a = 1.5(30) - 30 = 15$

$k = 4$ $x_1 = \min\{40/2, 100\} = 20$

$x_2 = \min\{60/1, 80\} = 60$

$x_3 = \min\{20/0.5, 20\} = \min\{40, 20\} = 20$

$x_a = \min\{30/0, 0\} = 0;\ p_a = 0;\ u_a = 1.5(0) - 0 = 0$

Game-Theoretic Approach (cont.)

Some questions

- What is the computational complexity of finding a best response as described, under the deterministic bid assumption?
- How might we generalize this approach to handle multiple market segments? (It seems making an assumption about other agents' budgets, as well as bids, could help.)
- What about the many possible different orderings of the users? How can we manage this uncertainty?
- How might the other agents best-respond to our agent's choice of k^*, and its corresponding bids?

Market Equilibrium Approach

High-Level Idea

Compute a competitive (or **Walrasian**) equilibrium of the market induced by the game, and then program your agent to **bid its part** of this equilibrium.

Market Equilibrium Approach

High-Level Idea

Compute a competitive (or **Walrasian**) equilibrium of the market induced by the game, and then program your agent to **bid its part** of this equilibrium.

Market Equilibrium Approach

High-Level Idea

Compute a competitive (or **Walrasian**) equilibrium of the market induced by the game, and then program your agent to **bid its part** of this equilibrium.

Definition: Walrasian equilibrium

An allocation and a pricing such that:

- All agents/buyers are utility maximizing.
- The market clears: i.e., supply meets demand:
  i.e., the seller is also revenue maximizing.

Market Equilibrium Approach (cont.)

In combinatorial markets, we can compute a Walrasian equilibrium with the following ILP. Here, the set of goods G comprises users.

$$\text{maximize}_{\boldsymbol{x}} \quad \sum_i \sum_{S \subseteq G} v_i(S)\, x_{iS}$$

$$\begin{aligned}
\text{subject to} \quad & \sum_{S \subseteq G} x_{iS} \leq 1, \quad \forall i \\
& \sum_i \sum_{S \subseteq G : k \in S} x_{iS} \leq 1, \quad \forall k \in G \\
& x_{iS} \in \{0, 1\}, \quad \forall i, \forall S \subseteq G
\end{aligned}$$

Market Equilibrium Approach (cont.)

Wait a minute!!! Where are the prices in this ILP!!!

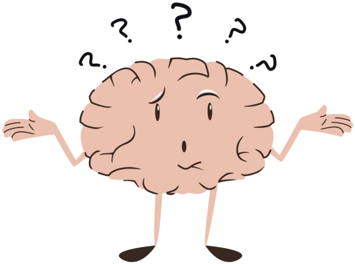


Figure: My brain is confused!

Market Equilibrium Approach (cont.)

Prices are in the dual of the ILP relaxation (i.e., the corresponding linear program, where x_{iS} can be fractional: i.e., $x_{iS} \in [0, 1]$)!

$$\text{minimize}_{\boldsymbol{u},\boldsymbol{p}} \quad \sum_i u_i + \sum_k p_k$$

$$\text{subject to} \quad u_i + \sum_{k \in S} p_k \leq 1, \quad \forall i, \forall S \subseteq G$$

$$u_i, p_k \geq 0, \quad \forall i, \forall k$$

Market Equilibrium Approach (cont.)

Great! But can we actually solve this ILP in a reasonable amount of time? Note:

Market Equilibrium Approach (cont.)

Great! But can we actually solve this ILP in a reasonable amount of time? Note:

- There are exponentially many variables: one per subset of users.

Market Equilibrium Approach (cont.)

Great! But can we actually solve this ILP in a reasonable amount of time? Note:

- There are exponentially many variables: one per subset of users.
- ILP quickly becomes intractable; we assume thousands of users.

Market Equilibrium Approach (cont.)

Great! But can we actually solve this ILP in a reasonable amount of time? Note:

- There are exponentially many variables: one per subset of users.
- ILP quickly becomes intractable; we assume thousands of users.
- Is solving the ILP in P? in NP?

Market Equilibrium Approach (cont.)

Great! But can we actually solve this ILP in a reasonable amount of time? Note:

- There are exponentially many variables: one per subset of users.
- ILP quickly becomes intractable; we assume thousands of users.
- Is solving the ILP in P? in NP?

WE computation for combinatorial markets

Answer to the last question: it depends on agents' valuations. (In P, for additive, unit demand, diminishing marginal values.)

Market Equilibrium Approach (cont.)

Question

How would you relax (i.e., what assumptions would you make) to be able to solve for a WE in a reasonable amount of time?

Market Equilibrium Approach (cont.)

Question

How would you relax (i.e., what assumptions would you make) to be able to solve for a WE in a reasonable amount of time?

Some remarks

- Note that WE computation is *independent* of agents' strategies!
- Most useful in a setting where other-agent strategies are difficult to predict.
- But what if we have reasonable predictions of other agents strategies?

Some General Remarks

Multiple days

The discussion so far has been for the one-day game.
Can you think of ways to generalize these ideas to multiple days?

Some General Remarks

Multiple days

The discussion so far has been for the one-day game.
Can you think of ways to generalize these ideas to multiple days?

Incomplete information

Going back to our assumptions, can we lift some of them?

Dealing with uncertainty

Given an approach for a complete-information setting (e.g., market or game-theoretic equilibria), how would you used it in the case of incomplete information?

Dealing with uncertainty

Given an approach for a complete-information setting (e.g., market or game-theoretic equilibria), how would you used it in the case of incomplete information?

Simple approach

- Sample the stochasticity in the game multiple times.

Dealing with uncertainty

Given an approach for a complete-information setting (e.g., market or game-theoretic equilibria), how would you used it in the case of incomplete information?

Simple approach

- Sample the stochasticity in the game multiple times.
- For each sample, compute a bid vector and limits.

Dealing with uncertainty

Given an approach for a complete-information setting (e.g., market or game-theoretic equilibria), how would you used it in the case of incomplete information?

Simple approach

- Sample the stochasticity in the game multiple times.
- For each sample, compute a bid vector and limits.
- Aggregate (e.g., average) these bid vectors and limits.

Dealing with uncertainty

Given an approach for a complete-information setting (e.g., market or game-theoretic equilibria), how would you used it in the case of incomplete information?

Simple approach

- Sample the stochasticity in the game multiple times.
- For each sample, compute a bid vector and limits.
- Aggregate (e.g., average) these bid vectors and limits.
- Report bids and limits based on the aggregate.

Two Equilibrium Approaches (Revisited)

Market Equilibrium Approach

- Ignore the strategic behavior of agents.
- But market equilibria do not always exist. May need to make simplifying assumptions (e.g., divisibility/Fisher markets).

Game-Theoretic Approach

- Assume strategic behavior on the part of the other agents. Calculate your agent's move as a best response.
- Iterative approach: **predict** (or **learn**) the strategic behavior of the other agents in the game, and then compute a best response to this prediction. Repeat. (May not converge.)