Learning Equilibria of Simulation-Based Games: Applications to Empirical Mechanism Design

Enrique Areyan Viqueira

March 12, 2021

Outline

- **Part 1**: Simulation-Based Games
- **Part 2**: Combinatorial Markets
- **Part 3**: Empirical Mechanism Design (if time permits)

Part 1:
Learning Equilibria of Simulation-Based Games

Improved Algorithms for Learning Equilibria in Simulation-Based Games.
Enrique Areyan Viqueira, Cyrus Cousins, Amy Greenwald.
19th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS20).

Learning Simulation-Based Games from Data.
Enrique Areyan Viqueira, Amy Greenwald, Cyrus Cousins, Eli Upfal.
18th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS19).

The "Game" Plan (a.k.a. Outline Part 1)

- Simulation-based Games
- Mathematical Framework
- Learning Algorithms
- Experimental Results

The "Game" Plan (a.k.a. Outline Part 1)

- Simulation-based Games
- Mathematical Framework
- Learning Algorithms
- Experimental Results

Simulation-Based Games

- **Game theory** is the standard conceptual framework to analyze the interaction among strategic agents

Simulation-Based Games

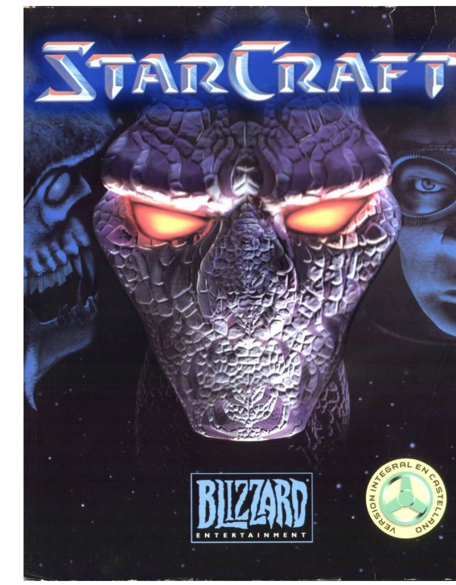
- **Game theory** is the standard conceptual framework to analyze the interaction among strategic agents

- At the heart of game theory is the notion of a **Game** - a mathematical object: players, actions, and utilities

Simulation-Based Games

- **Game theory** is the standard conceptual framework to analyze the interaction among strategic agents
- At the heart of game theory is the notion of a **Game** - a mathematical object: players, actions, and utilities
- Often, an analyst can specify a **game description** completely. But, there are games too complex to afford a complete description

Simulation-Based Games - Examples



- StarCraft: a real-time strategy game
- Hundreds of units and buildings, large strategy space
- Deepmind[1] recently built the first AI to defeat a top player

  Their parameterization of the game has an average of 10^{26} legal actions at each step!

[1] https://deepmind.com/blog/article/alphastar-mastering-real-time-strategy-game-starcraft-ii

Simulation-Based Games - Pervasive in Real Life

- As fun as StarCraft might be, think of it as a toy model for important, real-world applications of multi-agent systems such as:

  Electronic advertisement (TAC AdX - https://sites.google.com/site/gameadx/)

  Energy markets (Power TAC - https://powertac.org/)

  Industrial supply chains (ANAC-SCML http://web.tuat.ac.jp/~katfuji/ANAC2019/#scm)

  etc.

Simulation-Based Games - Characteristics

- Games are **too complex** to exactly compute expected utilities

Simulation-Based Games - Characteristics

- Games are **too complex** to exactly compute expected utilities
- Many sources of complexity, in the StarCraft example
  different terrains, units, actions, etc.

Simulation-Based Games - Characteristics

- Games are **too complex** to exactly compute expected utilities
- Many sources of complexity, in the StarCraft example
  different terrains, units, actions, etc.
- Nevertheless, in **simulation-based games**, one can obtain samples of utilities by running a **game simulator**

Simulation-Based Games - Mathematical Model

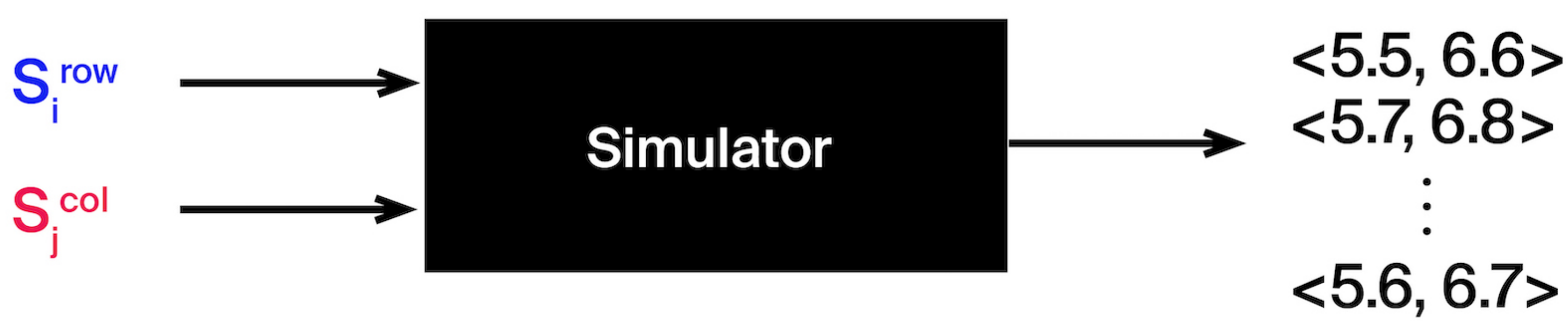
	S_1^{col}	S_2^{col}	$\cdots$	S_n^{col}
S_1^{row}	?, ?	?, ?	$\cdots$	?, ?
S_2^{row}	?, ?	?, ?	$\cdots$	?, ?
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
S_m^{row}	?, ?	?, ?	$\cdots$	?, ?

Simulation-based game

Simulation-Based Games - Mathematical Model

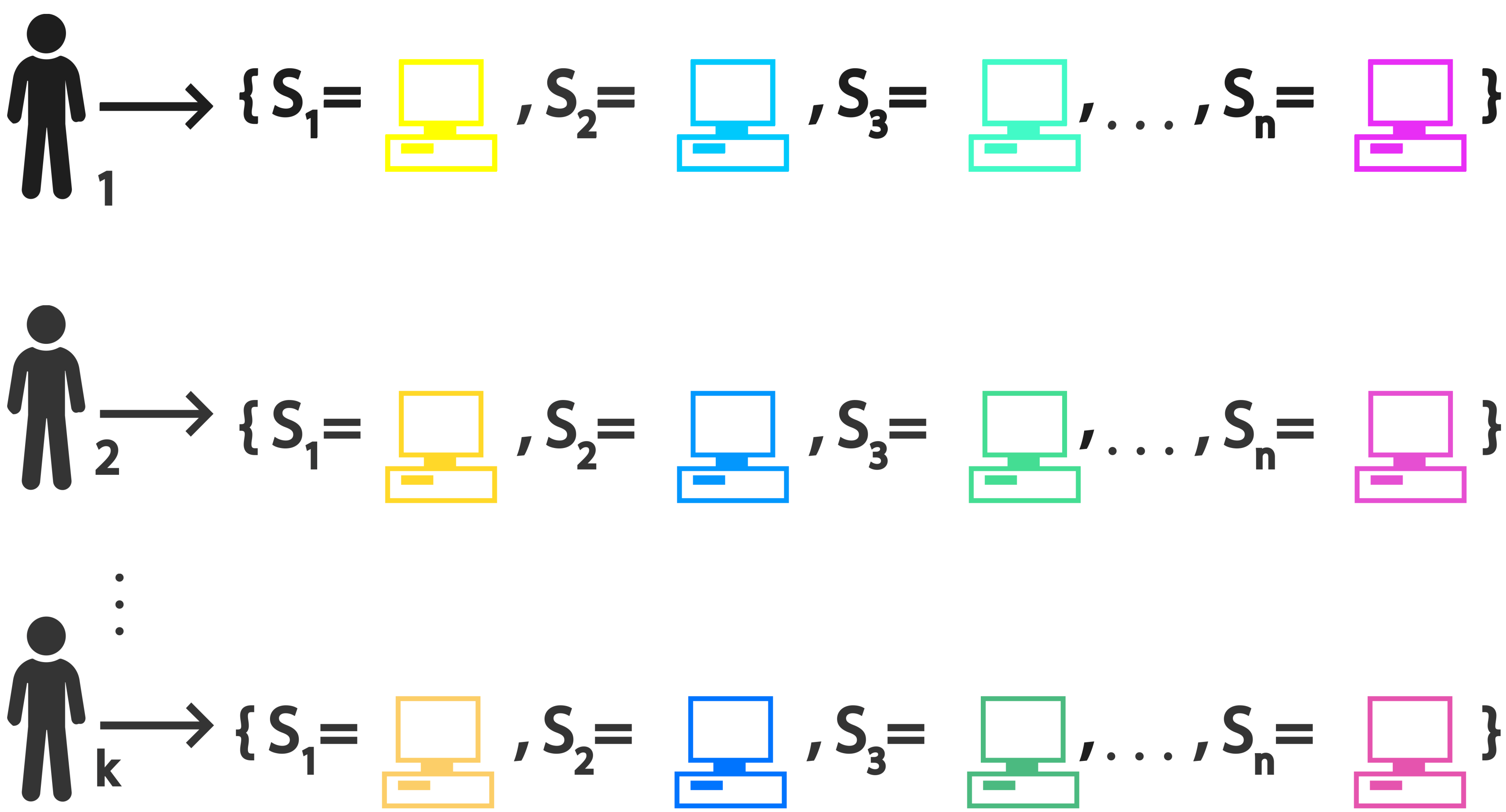
	S_1^{col}	S_2^{col}	$\cdots$	S_n^{col}
S_1^{row}	?, ?	?, ?	$\cdots$	?, ?
S_2^{row}	?, ?	?, ?	$\cdots$	?, ?
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
S_m^{row}	?, ?	?, ?	$\cdots$	?, ?

Simulation-based game



Simulation-Based Games - Heuristics

- Actions spaces are vast, so usually no optimal strategies are available. Instead, there are a few heuristics.



Plan for the rest of Part 1

- High-level Goal: learn the equilibria of simulation-based games
- Formalize simulation-based games and their equilibria
- Learning algorithms and experimental results

The "Game" Plan (a.k.a. Outline Part 1)

- Simulation-based Games
- Mathematical Framework
- Learning Algorithms
- Experimental Results

The "Game" Plan (a.k.a. Outline Part 1)

- ~~Simulation-based Games~~
- Mathematical Framework
- Learning Algorithms
- Experimental Results

A Mathematical Model - Conditional and Expected Games

$\vec{s} = (s_1, s_2, \ldots, s_n)$, where s_i is agent's i strategy

- Let $u_p(\vec{s})$ be player p's utility when **strategy profile** $\vec{s}$ is played

A Mathematical Model - Conditional and Expected Games

$\vec{s} = (s_1, s_2, \ldots, s_n)$, where s_i is agent's i strategy

- Let $u_p(\vec{s})$ be player p's utility when **strategy profile** $\vec{s}$ is played
- Model **randomness** by postulating a set of **conditions** $\mathcal{X}$, such that given $x \in \mathcal{X}$, we obtain a utility function $u_p(\vec{s}; x)$

A Mathematical Model - Conditional and Expected Games

$\vec{s} = (s_1, s_2, \ldots, s_n)$, where s_i is agent's i strategy

- Let $u_p(\vec{s})$ be player p's utility when **strategy profile** $\vec{s}$ is played

- Model **randomness** by postulating a set of **conditions** $\mathcal{X}$, such that given $x \in \mathcal{X}$, we obtain a utility function $u_p(\vec{s}; x)$

- Given a distribution $\mathcal{D}$ over condition set $\mathcal{X}$, we define the **expected** utility $\bar{u}_p(\vec{s}) = \mathbb{E}_{x \sim \mathcal{D}}[u_p(\vec{s}; x)]$

A Mathematical Model - Conditional and Expected Games

$\vec{s} = (s_1, s_2, \ldots, s_n)$, where s_i is agent's i strategy

- Let $u_p(\vec{s})$ be player p's utility when **strategy profile** $\vec{s}$ is played

- Model **randomness** by postulating a set of **conditions** $\mathscr{X}$, such that given $x \in \mathscr{X}$, we obtain a utility function $u_p(\vec{s}; x)$

- Given a distribution $\mathscr{D}$ over condition set $\mathscr{X}$, we define the **expected** utility $\bar{u}_p(\vec{s}) = \mathbb{E}_{x \sim \mathscr{D}}[u_p(\vec{s}; x)]$

- The **expected game** (the normal-form game with expected utilities) is then our model of a simulation-based game

A Mathematical Model - Empirical Games

- Recall that, in practice, we only observe **samples** of the utilities of simulation-based games

A Mathematical Model - Empirical Games

- Recall that, in practice, we only observe **samples** of the utilities of simulation-based games

- Given m samples: $u_p(\vec{s}; x_1), u_p(\vec{s}; x_2), \cdots, u_p(\vec{s}; x_m)$
  The **empirical utility** is the average: $\hat{u}_p(\vec{s}) = \frac{1}{m} \sum_{i=1}^{m} u_p(\vec{s}; x_i)$

A Mathematical Model - Empirical Games

- Recall that, in practice, we only observe **samples** of the utilities of simulation-based games

- Given m samples: $u_p(\vec{s}; x_1), u_p(\vec{s}; x_2), \cdots, u_p(\vec{s}; x_m)$
  The **empirical utility** is the average: $\hat{u}_p(\vec{s}) = \frac{1}{m} \sum_{i=1}^{m} u_p(\vec{s}; x_i)$

- The **empirical game** has empirical utilities for every player and strategy profile

Goal

Learn, with provable guarantees, **all** the **equilibria** of **expected games** given access only to **empirical games**

(Other valid and interesting goals:

+ recover one equilibrium, e.g., by following best-response dynamics)

Approximation Framework

- Our mathematical tool: ϵ-uniform approximations (here $\varepsilon > 0$).

Approximation Framework

- Our mathematical tool: ϵ-uniform approximations (here $\varepsilon > 0$).
- A game G_1 is an ϵ-**uniform approximation** of game G_2 if

Approximation Framework

- Our mathematical tool: ϵ-uniform approximations (here $\varepsilon > 0$).
- A game G_1 is an ϵ-**uniform approximation** of game G_2 if

$$\forall p, \vec{s} : |u_p^1(\vec{s}) - u_p^2(\vec{s})| < \varepsilon$$

Approximation Framework

- Our mathematical tool: ϵ-uniform approximations (here $\varepsilon > 0$).
- A game G_1 is an ϵ-**uniform approximation** of game G_2 if

$$\forall p, \vec{s} : |u_p^1(\vec{s}) - u_p^2(\vec{s})| < \varepsilon$$

G_1	S_1^{col}	S_2^{col}
S_1^{row}	-1, -1	-3, 0
S_2^{row}	0, -3	-2, -2

G_2	S_1^{col}	S_2^{col}
S_1^{row}	-1, -1	-3, 0 + ε
S_2^{row}	0, -3 - ε	-2 + $\frac{\varepsilon}{4}$, -2

Approximation Framework

- Our mathematical tool: ϵ-uniform approximations (here $\varepsilon > 0$).
- A game G_1 is an ϵ-**uniform approximation** of game G_2 if

$$\forall p, \vec{s} : |u_p^1(\vec{s}) - u_p^2(\vec{s})| < \varepsilon$$

G_1	S_1^{col}	S_2^{col}
S_1^{row}	-1, -1	-3, 0
S_2^{row}	0, -3	-2, -2

G_2	S_1^{col}	S_2^{col}
S_1^{row}	-1, -1	-3, 0 $+\varepsilon$
S_2^{row}	0, -3 $-\varepsilon$	-2 $+\frac{\varepsilon}{4}$, -2

Approximation Framework

- Our mathematical tool: ϵ-uniform approximations (here $\varepsilon > 0$).
- A game G_1 is an ϵ-**uniform approximation** of game G_2 if

$$\forall p, \vec{s} : |u_p^1(\vec{s}) - u_p^2(\vec{s})| < \varepsilon$$

G_1 **is** ε**-close to** G_2

G_1	S_1^{col}	S_2^{col}
S_1^{row}	-1, -1	-3, 0
S_2^{row}	0, -3	-2, -2

G_2	S_1^{col}	S_2^{col}
S_1^{row}	-1, -1	-3, 0 $+\varepsilon$
S_2^{row}	0, -3 $-\varepsilon$	-2 $+\frac{\varepsilon}{4}$, -2

Approximation Framework

- For simulation-based games, $\mathscr{X}$, and $\mathscr{D}$ are complex objects. We can't reasonably hope to compute the expected game exactly

Approximation Framework

- For simulation-based games, $\mathscr{X}$, and $\mathscr{D}$ are complex objects. We can't reasonably hope to compute the expected game exactly
- Even if we could approximate each $\bar{u}_p(\vec{s})$ (say, up to ε), would that destroy the equilibria?

Approximation Framework

- For simulation-based games, $\mathscr{X}$, and $\mathscr{D}$ are complex objects. We can't reasonably hope to compute the expected game exactly

- Even if we could approximate each $\bar{u}_p(\vec{s})$ (say, up to ε), would that destroy the equilibria?

- **Definition**: a strategy profile $\vec{s}$ is an ε-**equilibrium** if players don't have incentive to deviate, up to ε, fixing other players' strategies

Approximating Equilibria - First Result

Theorem: (Recall-Precision)

Approximating Equilibria - First Result

Theorem: (Recall-Precision)

- If G_1 is an ϵ-**uniform approximation** of game G_2, then

Approximating Equilibria - First Result

Theorem: (Recall-Precision)

- If G_1 is an ϵ-**uniform approximation** of game G_2, then
  - **Recall**: Every equilibrium of G_1 is a 2ε-equilibrium of G_2

Approximating Equilibria - First Result

Theorem: (Recall-Precision)

- If G_1 is an ϵ-**uniform approximation** of game G_2, then
  - **Recall**: Every equilibrium of G_1 is a 2ε-equilibrium of G_2
  - **Precision**: Every 2ε-equilibrium of G_2 is a 4ε-equilibrium of G_1

Approximating Equilibria - First Result

Tuyls, K. et al. Bounds and dynamics for empirical game theoretic analysis, 2020.

Theorem: (Recall-Precision)

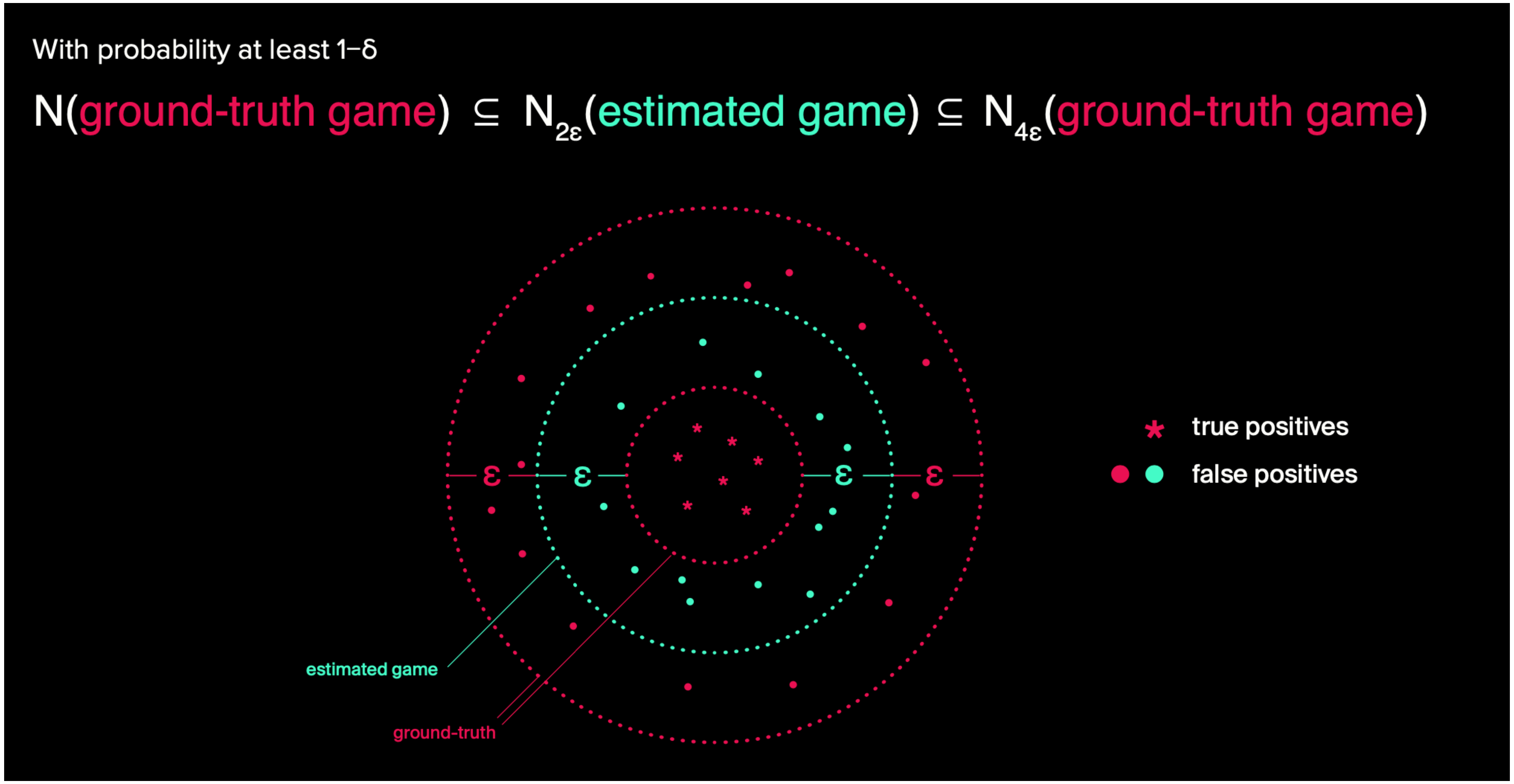
- If G_1 is an ϵ-**uniform approximation** of game G_2, then
  - **Recall**: Every equilibrium of G_1 is a 2ε-equilibrium of G_2
  - **Precision**: Every 2ε-equilibrium of G_2 is a 4ε-equilibrium of G_1

Approximating Equilibria - First Result

Tuyls, K. et al. Bounds and dynamics for empirical game theoretic analysis, 2020.

Theorem: (Recall-Precision)

- If G_1 is an ϵ-**uniform approximation** of game G_2, then
  - **Recall**: Every equilibrium of G_1 is a 2ε-equilibrium of G_2
  - **Precision**: Every 2ε-equilibrium of G_2 is a 4ε-equilibrium of G_1



Learning Equilibria

How to learn the approximate equilibria
of a simulation-based game from sample data?

Original Goal

How to learn an ϵ-uniform approximation of
an expected game from sample data?

Mathematically Precise Goal

The "Game" Plan (a.k.a. Outline Part 1)

- ~~Simulation-based Games~~
- Mathematical Framework
- Learning Algorithms
- Experimental Results

The "Game" Plan (a.k.a. Outline Part 1)

- ~~Simulation-based Games~~
- ~~Mathematical Framework~~
- Learning Algorithms
- Experimental Results

Learning Algorithms - A Baseline

- We present two **Probably Approximate Correct** (PAC) algorithm to learn empirical games

Learning Algorithms - A Baseline

- We present two **Probably Approximate Correct** (PAC) algorithm to learn empirical games
  - **PAC** algorithm: given $\varepsilon, \delta > 0$, learn some model (games!) up to error at most ε and with probability at least $1 - \delta$

Learning Algorithms - A Baseline

How much error are you willing to tolerate?
Error tolerance ε

- We present two **Probably Approximate Correct** (PAC) algorithm to learn empirical games

  - **PAC** algorithm: given $\varepsilon, \delta > 0$, learn some model (games!) up to error at most ε and with probability at least $1 - \delta$

Learning Algorithms - A Baseline

How much error are you willing to tolerate?
Error tolerance ε

How likely would you like the estimates to be?
Failure tolerance δ

- We present two **Probably Approximate Correct** (PAC) algorithm to learn empirical games

  - **PAC** algorithm: given $\varepsilon, \delta > 0$, learn some model (games!) up to error at most ε and with probability at least $1 - \delta$

Learning Algorithms - A Baseline

How much error are you willing to tolerate?
Error tolerance ε

How likely would you like the estimates to be?
Failure tolerance δ

- We present two **Probably Approximate Correct** (PAC) algorithm to learn empirical games
  - **PAC** algorithm: given $\varepsilon, \delta > 0$, learn some model (games!) up to error at most ε and with probability at least $1 - \delta$
- The first algorithm is a baseline that uses **Hoeffding's Inequality** to estimate all utilities of a simulation-based game

Tuyls, K. et al.
Bounds and dynamics for empirical game theoretic analysis, 2020.

Learning Algorithms - Progressive Sampling with Pruning

- Recall our goal: learn **equilibria.** Not all utilities we learn are relevant to get at equilibria. For example,

Learning Algorithms - Progressive Sampling with Pruning

- Recall our goal: learn **equilibria.** Not all utilities we learn are relevant to get at equilibria. For example,

	S_1^{col}	S_2^{col}
S_1^{row}	3, 3	0, 5
S_2^{row}	5, 0	2, 2

What value of ε is enough to estimate the equilibrium of this game?

Learning Algorithms - Progressive Sampling with Pruning

- Recall our goal: learn **equilibria.** Not all utilities we learn are relevant to get at equilibria. For example,

	S_1^{col}	S_2^{col}
S_1^{row}	3, 3	0, 5
S_2^{row}	5, 0	2, 2

We don't need to learn "3" exactly, we just need to learn that "5">"3", up to errors.

What value of ε is enough to estimate the equilibrium of this game?

Learning Algorithms - Progressive Sampling with Pruning

- Recall our goal: learn **equilibria.** Not all utilities we learn are relevant to get at equilibria. For example,

	S_1^{col}	S_2^{col}
S_1^{row}	3, 3	0, 5
S_2^{row}	5, 0	2, 2

We don't need to learn "3" exactly, we just need to learn that "5">"3", up to errors.

What value of ε is enough to estimate the equilibrium of this game?

$$\varepsilon < 1$$

Learning Algorithms - Progressive Sampling with Pruning

- Recall our goal: learn **equilibria.** Not all utilities we learn are relevant to get at equilibria. For example,

	S_1^{col}	S_2^{col}
S_1^{row}	3, 3	0, 5
S_2^{row}	5, 0	2, 2

We don't need to learn "3" exactly, we just need to learn that "5">"3", up to errors.

What value of ε is enough to estimate the equilibrium of this game?

$$\varepsilon < 1$$

- **Idea**: take a few samples first, then take more samples of only those profiles that can't be refuted as part of an equilibrium

Learning Algorithms - Progressive Sampling with Pruning

- **Algorithm:** Progressive Sampling With Pruning

Learning Algorithms - Progressive Sampling with Pruning

- **Algorithm:** Progressive Sampling With Pruning
  - Initially, all $p, \vec{s}$ are active. Initial error ϵ_0 is "big".

Learning Algorithms - Progressive Sampling with Pruning

- **Algorithm:** Progressive Sampling With Pruning
  - Initially, all $p, \vec{s}$ are active. Initial error ϵ_0 is "big".
  - While some target accuracy ϵ is not reached ($\epsilon < \epsilon_t$) or

    we run out of sampling budget or

    there are no more active $p, \vec{s}$

Learning Algorithms - Progressive Sampling with Pruning

- **Algorithm:** Progressive Sampling With Pruning
  - Initially, all $p, \vec{s}$ are active. Initial error ϵ_0 is "big".
  - While some target accuracy ϵ is not reached ($\epsilon < \epsilon_t$) or

    we run out of sampling budget or

    there are no more active $p, \vec{s}$

    - Sample all active $p, \vec{s}$, up to current error ϵ_t

Learning Algorithms - Progressive Sampling with Pruning

- **Algorithm:** Progressive Sampling With Pruning
  - Initially, all $p, \vec{s}$ are active. Initial error ϵ_0 is "big".
  - While some target accuracy ϵ is not reached ($\epsilon < \epsilon_t$) or

    we run out of sampling budget or

    there are no more active $p, \vec{s}$

    - Sample all active $p, \vec{s}$, up to current error ϵ_t
    - For all active $\vec{s}$
      - if $\vec{s}$ can be refuted as part of an equilibrium,

        then remove it from the active set

Learning Algorithms - Progressive Sampling with Pruning

- **Algorithm:** Progressive Sampling With Pruning
  - Initially, all $p, \vec{s}$ are active. Initial error ϵ_0 is "big".
  - While some target accuracy ϵ is not reached ($\epsilon < \epsilon_t$) or
    we run out of sampling budget or
    there are no more active $p, \vec{s}$
    - Sample all active $p, \vec{s}$, up to current error ϵ_t
    - For all active $\vec{s}$
      - if $\vec{s}$ can be refuted as part of an equilibrium,
        then remove it from the active set
    - Decrease the target error $\epsilon_{t+1} \leftarrow \epsilon_t - \text{constant}$

The "Game" Plan (a.k.a. Outline Part 1)

- ~~Simulation-based Games~~
- ~~Mathematical Framework~~
- Learning Algorithms
- Experimental Results

The "Game" Plan (a.k.a. Outline Part 1)

- ~~Simulation-based Games~~
- ~~Mathematical Framework~~
- ~~Learning Algorithms~~
- Experimental Results

Experimental Setup

- We use GAMUT (gamut.stanford.edu) to generate games
- We use Gambit (www.gambit-project.org) for equilibria computation
- We developed a python library (github.com/eareyan/pysegta) that implements our learning algorithms and interfaces with both GAMUT and Gambit.

Experimental Results - Summary

- Pruning is highly data efficient over a wide range of games, specifically, over 10 different classes of games

Experimental Results - Summary

- Pruning is highly data efficient over a wide range of games, specifically, over 10 different classes of games
- Efficiency is due to the algorithm exploiting the strategic structure of games without knowing *a priori* what this structure is!

Experimental Results - Summary

	$\varepsilon \leq 0.125$		$\varepsilon \leq 0.25$		$\varepsilon \leq 0.5$		$\varepsilon \leq 1.0$	
Bound	Hoeffding	Emp. Bennett	Hoeffding	Emp. Bennett	Hoeffding	Emp. Bennett	Hoeffding	Emp. Bennett
Game/Algorithm	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}
Congestion Games (5 facilities)	3,051; **1,654**; 0.08	3,051; **1,449**; 0.00	762; **464**; 0.17	762; **364**; 0.01	190; **146**; 0.34	190; **93**; 0.01	**47**; 58; 0.70	47; **25**; 0.04
Zero-Sum Games (30 strategies)	2,841; **1,691**; 0.08	2,841; **1,383**; 0.00	710; **502**; 0.17	710; **349**; 0.01	177; **166**; 0.35	177; **90**; 0.01	**44**; 62; 0.71	44; **25**; 0.04
Random Games (30 strategies)	2,841; **1,666**; 0.08	2,841; **1,375**; 0.00	710; **491**; 0.17	710; **347**; 0.01	177; **159**; 0.35	177; **90**; 0.01	**44**; 58; 0.71	44; **25**; 0.04
Congestion Games (4 facilities)	622; **492**; 0.09	622; **438**; 0.00	156; **138**; 0.17	156; **110**; 0.01	**39**; 41; 0.35	39; **28**; 0.01	**10**; 15; 0.71	10; **8**; 0.04
Zero-Sum Games (20 strategies)	1,171; **829**; 0.09	1,171; **708**; 0.00	293; **240**; 0.17	293; **179**; 0.01	**73**; 77; 0.35	73; **46**; 0.01	**18**; 28; 0.71	18; **13**; 0.04
Random Games (20 strategies)	1,171; **809**; 0.09	1,171; **698**; 0.00	293; **232**; 0.17	293; **176**; 0.01	**73**; **73**; 0.35	73; **45**; 0.01	**18**; 25; 0.71	18; **12**; 0.04
Congestion Games (3 facilities)	**114**; 145; 0.09	**114**; 135; 0.00	**29**; 40; 0.18	**29**; 34; 0.01	**7**; 12; 0.36	**7**; 9; 0.02	**2**; 4; 0.73	**2**; **2**; 0.05
Zero-Sum Games (10 strategies)	**254**; 268; 0.09	254; **242**; 0.00	**63**; 73; 0.18	63; **61**; 0.01	**16**; 22;0.36	16; **15**;0.02	**4**; 7; 0.73	**4**; **4**; 0.05
Random Games (10 strategies)	**254**; **254**; 0.09	254; **233**; 0.00	**63**; 69; 0.18	63; **59**; 0.01	**16**; 21;0.36	16; **15**; 0.02	**4**; 7; 0.72	**4**; **4**; 0.05
Congestion Games (2 facilities)	**17**; 37; 0.09	**17**; 37; 0.00	**4**; 10; 0.19	**4**; 9; 0.01	**1**; 3; 0.38	**1**; 2; 0.02	**1**; **1**; 0.76	**1**; **1**; 0.05
Zero-Sum Games (5 strategies)	**54**; 94; 0.09	**54**; 89; 0.00	**13**; 25; 0.18	**13**; 22; 0.01	**3**; 7; 0.37	**3**; 6; 0.02	**1**; 2; 0.75	**1**; **1**; 0.05
Random Games (5 strategies)	**54**; 83; 0.09	**54**; 90; 0.00	**13**; 22; 0.18	**13**; 20; 0.01	**3**; 6; 0.37	**3**; 5; 0.02	**1**; 2; 0.74	**1**; **1**; 0.05

Table 1: PSP's **sample efficiency. Numbers of samples are reported in tens of thousands. The values in bold are smaller than their counterparts; as ε is fixed, they indicate the more sample efficient algorithms.**

Experimental Results - Summary

	$\varepsilon \leq 0.125$		$\varepsilon \leq 0.25$		$\varepsilon \leq 0.5$		$\varepsilon \leq 1.0$	
Bound	Hoeffding	Emp. Bennett	Hoeffding	Emp. Bennett	Hoeffding	Emp. Bennett	Hoeffding	Emp. Bennett
Game/Algorithm	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}
Congestion Games (5 facilities)	3,051; **1,654**; 0.08	3,051; **1,449**; 0.00	762; **464**; 0.17	762; **364**; 0.01	190; **146**; 0.34	190; **93**; 0.01	**47**; 58; 0.70	47; **25**; 0.04
Zero-Sum Games (30 strategies)	2,841; **1,691**; 0.08	2,841; **1,383**; 0.00	710; **502**; 0.17	710; **349**; 0.01	177; **166**; 0.35	177; **90**; 0.01	**44**; 62; 0.71	44; **25**; 0.04
Random Games (30 strategies)	2,841; **1,666**; 0.08	2,841; **1,375**; 0.00	710; **491**; 0.17	710; **347**; 0.01	177; **159**; 0.35	177; **90**; 0.01	**44**; 58; 0.71	44; **25**; 0.04
Congestion Games (4 facilities)	622; **492**; 0.09	622; **438**; 0.00	156; **138**; 0.17	156; **110**; 0.01	**39**; 41; 0.35	39; **28**; 0.01	**10**; 15; 0.71	10; **8**; 0.04
Zero-Sum Games (20 strategies)	1,171; **829**; 0.09	1,171; **708**; 0.00	293; **240**; 0.17	293; **179**; 0.01	**73**; 77; 0.35	73; **46**; 0.01	**18**; 28; 0.71	18; **13**; 0.04
Random Games (20 strategies)	1,171; **809**; 0.09	1,171; **698**; 0.00	293; **232**; 0.17	293; **176**; 0.01	**73**; **73**; 0.35	73; **45**; 0.01	**18**; 25; 0.71	18; **12**; 0.04
Congestion Games (3 facilities)	**114**; 145; 0.09	**114**; 135; 0.00	**29**; 40; 0.18	**29**; 34; 0.01	**7**; 12; 0.36	**7**; 9; 0.02	**2**; 4; 0.73	**2**; **2**; 0.05
Zero-Sum Games (10 strategies)	**254**; 268; 0.09	254; **242**; 0.00	**63**; 73; 0.18	63; **61**; 0.01	**16**; 22;0.36	16; **15**;0.02	**4**; 7; 0.73	**4**; **4**; 0.05
Random Games (10 strategies)	**254**; **254**; 0.09	254; **233**; 0.00	**63**; 69; 0.18	63; **59**; 0.01	**16**; 21;0.36	16; **15**; 0.02	**4**; 7; 0.72	**4**; **4**; 0.05
Congestion Games (2 facilities)	**17**; 37; 0.09	**17**; 37; 0.00	**4**; 10; 0.19	**4**; 9; 0.01	**1**; 3; 0.38	**1**; 2; 0.02	**1**; **1**; 0.76	**1**; **1**; 0.05
Zero-Sum Games (5 strategies)	**54**; 94; 0.09	**54**; 89; 0.00	**13**; 25; 0.18	**13**; 22; 0.01	**3**; 7; 0.37	**3**; 6; 0.02	**1**; 2; 0.75	**1**; **1**; 0.05
Random Games (5 strategies)	**54**; 83; 0.09	**54**; 90; 0.00	**13**; 22; 0.18	**13**; 20; 0.01	**3**; 6; 0.37	**3**; 5; 0.02	**1**; 2; 0.74	**1**; **1**; 0.05

Table 1: PSP's **sample efficiency. Numbers of samples are reported in tens of thousands. The values in bold are smaller than their counterparts; as ε is fixed, they indicate the more sample efficient algorithms.**

Experimental Results - Summary

Bound	$\varepsilon \leq 0.125$ Hoeffding	Emp. Bennett
Game/Algorithm	GS; PSP; ε_{PSP}	GS; PSP; ε_{PSP}
Congestion Games (5 facilities)	3,051; **1,654**; 0.08	3,051; **1,449**; 0.00
Zero-Sum Games (30 strategies)	2,841; **1,691**; 0.08	2,841; **1,383**; 0.00
Random Games (30 strategies)	2,841; **1,666**; 0.08	2,841; **1,375**; 0.00
Congestion Games (4 facilities)	622; **492**; 0.09	622; **438**; 0.00
Zero-Sum Games (20 strategies)	1,171; **829**; 0.09	1,171; **708**; 0.00
Random Games (20 strategies)	1,171; **809**; 0.09	1,171; **698**; 0.00
Congestion Games (3 facilities)	**114**; 145; 0.09	**114**; 135; 0.00
Zero-Sum Games (10 strategies)	**254**; 268; 0.09	254; **242**; 0.00
Random Games (10 strategies)	**254**; **254**; 0.09	254; **233**; 0.00
Congestion Games (2 facilities)	**17**; 37; 0.09	**17**; 37; 0.00
Zero-Sum Games (5 strategies)	**54**; 94; 0.09	**54**; 89; 0.00
Random Games (5 strategies)	**54**; 83; 0.09	**54**; 90; 0.00

Gam
Congestion Games
Zero-Sum Games (3
Random Games (3
Congestion Games
Zero-Sum Games (2
Random Games (2
Congestion Games
Zero-Sum Games (1
Random Games (1
Congestion Games
Zero-Sum Games (
Random Games (

Table 1: PSP's sa
their counterpa

$\varepsilon \leq 1.0$ ng	Emp. Bennett
SP	GS; PSP; ε_{PSP}
70	47; **25**; 0.04
71	44; **25**; 0.04
71	44; **25**; 0.04
71	10; **8**; 0.04
71	18; **13**; 0.04
71	18; **12**; 0.04
73	**2**; **2**; 0.05
73	**4**; **4**; 0.05
72	**4**; **4**; 0.05
76	**1**; **1**; 0.05
75	**1**; **1**; 0.05
74	**1**; **1**; 0.05

e smaller than

The "Game" Plan (a.k.a. Outline Part 1)

- ~~Simulation-based Games~~
- ~~Mathematical Framework~~
- ~~Learning Algorithms~~
- Experimental Results

The "Game" Plan (a.k.a. Outline Part 1)

- ~~Simulation-based Games~~
- ~~Mathematical Framework~~
- ~~Learning Algorithms~~
- ~~Experimental Results~~

Summary Part 1

- We contribute an end-to-end methodology for the analysis of simulation-based games

Summary Part 1

- We contribute an end-to-end methodology for the analysis of simulation-based games

- We prove tight bounds on the set of approximate equilibria of games learned from data

Summary Part 1

- We contribute an end-to-end methodology for the analysis of simulation-based games
- We prove tight bounds on the set of approximate equilibria of games learned from data
- We present and empirically evaluate a learning algorithm that exploits strategic structure of games to save on samples

Summary Part 1

- We contribute an end-to-end methodology for the analysis of simulation-based games
- We prove tight bounds on the set of approximate equilibria of games learned from data
- We present and empirically evaluate a learning algorithm that exploits strategic structure of games to save on samples
- We contribute an open-source library that implements our learning algorithms www.github.com/eareyan/pysegta

Part 2:
Learning Competitive Equilibria in Combinatorial Markets

Learning Competitive Equilibria in Noisy Combinatorial Markets
Enrique Areyan Viqueira, Cyrus Cousins, Amy Greenwald.
20th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS21).

Outline - Combinatorial Markets

- Model and Examples
- Noisy Combinatorial Markets
- Revisiting Pruning and Experiments

Combinatorial Markets

- Markets with **indivisible** goods

Combinatorial Markets

- Markets with **indivisible** goods
- Buyers can have **complex preferences** over bundles of goods

Combinatorial Markets

- Markets with **indivisible** goods
- Buyers can have **complex preferences** over bundles of goods
- They can be very economically **efficient**:

Combinatorial Markets

- Markets with **indivisible** goods
- Buyers can have **complex preferences** over bundles of goods
- They can be very economically **efficient**:
  - Flexibility to report complex preferences over a wide variety of outcomes might uncover value otherwise hidden

Model

Set of indivisible **goods**, G

Image credits: http://clipart-library.com/

Model

Set of indivisible **goods**, G

Bundle, $S \subseteq G$

Image credits: http://clipart-library.com/

Model

Set of indivisible **goods**, G

Set of **buyers**, N

Anna

Bob

Image credits: http://clipart-library.com/

Model

Set of indivisible **goods**, G

buyer $i \in N$

Anna

Bob

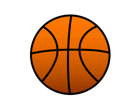
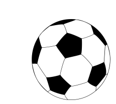
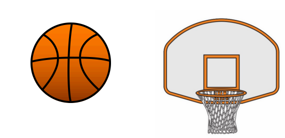
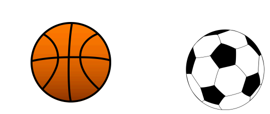
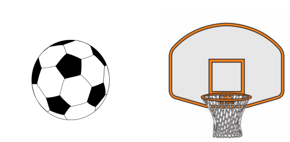
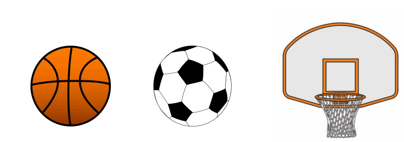
Bundle S	$v_i(S)$
$\emptyset$	**0**
	**1**
	**0**
	**1**
	**10**
	**2**
	**1**
	**11**

Image credits: http://clipart-library.com/

Model

Set of indivisible **goods**, G

Set of **buyers**, N

Anna

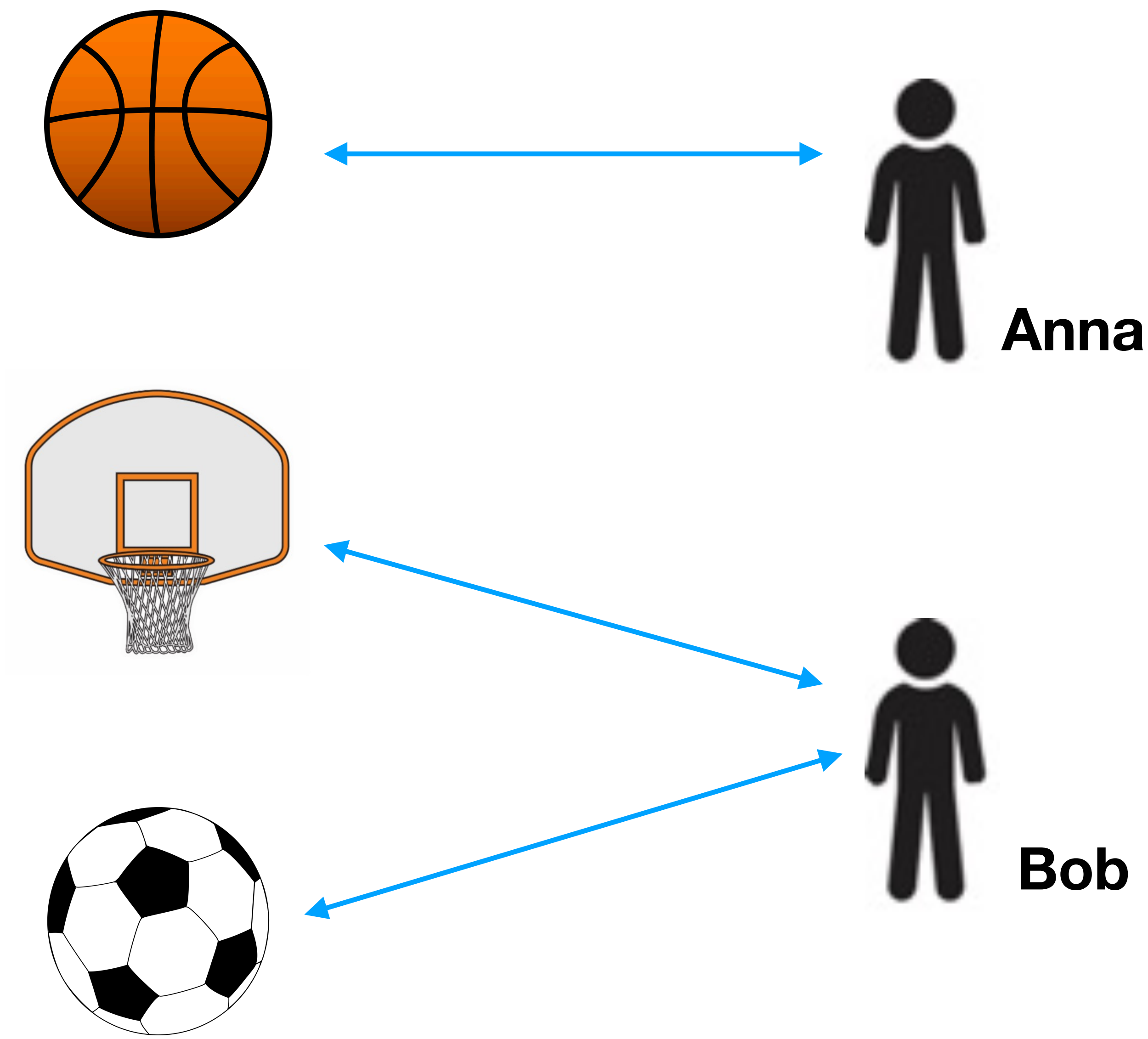
Bob

Image credits: http://clipart-library.com/

Model

Set of indivisible **goods**, G

Set of **buyers**, N



Allocation, $\mathcal{S} = (S_1, \ldots, S_n)$

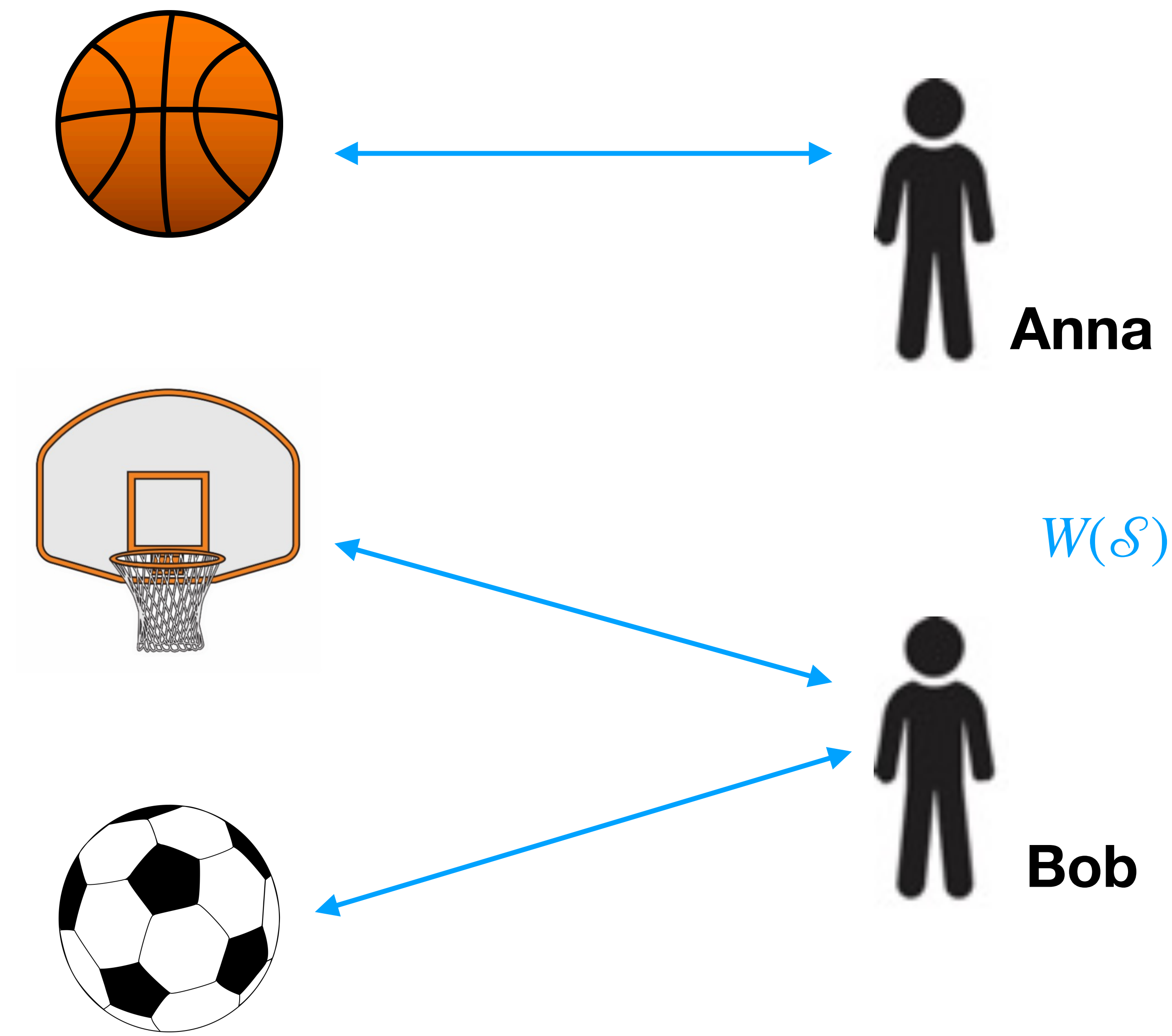
$S_i \cap S_j = \emptyset$

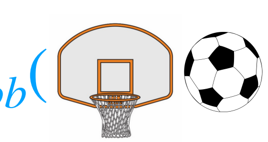
Image credits: http://clipart-library.com/

Model

Set of indivisible **goods**, G

Set of **buyers**, N



$W(\mathcal{S}) = v_{anna}($  $) + v_{bob}($  $)$

$= 1 + 1$

Allocation, $\mathcal{S} = (S_1, \ldots, S_n)$

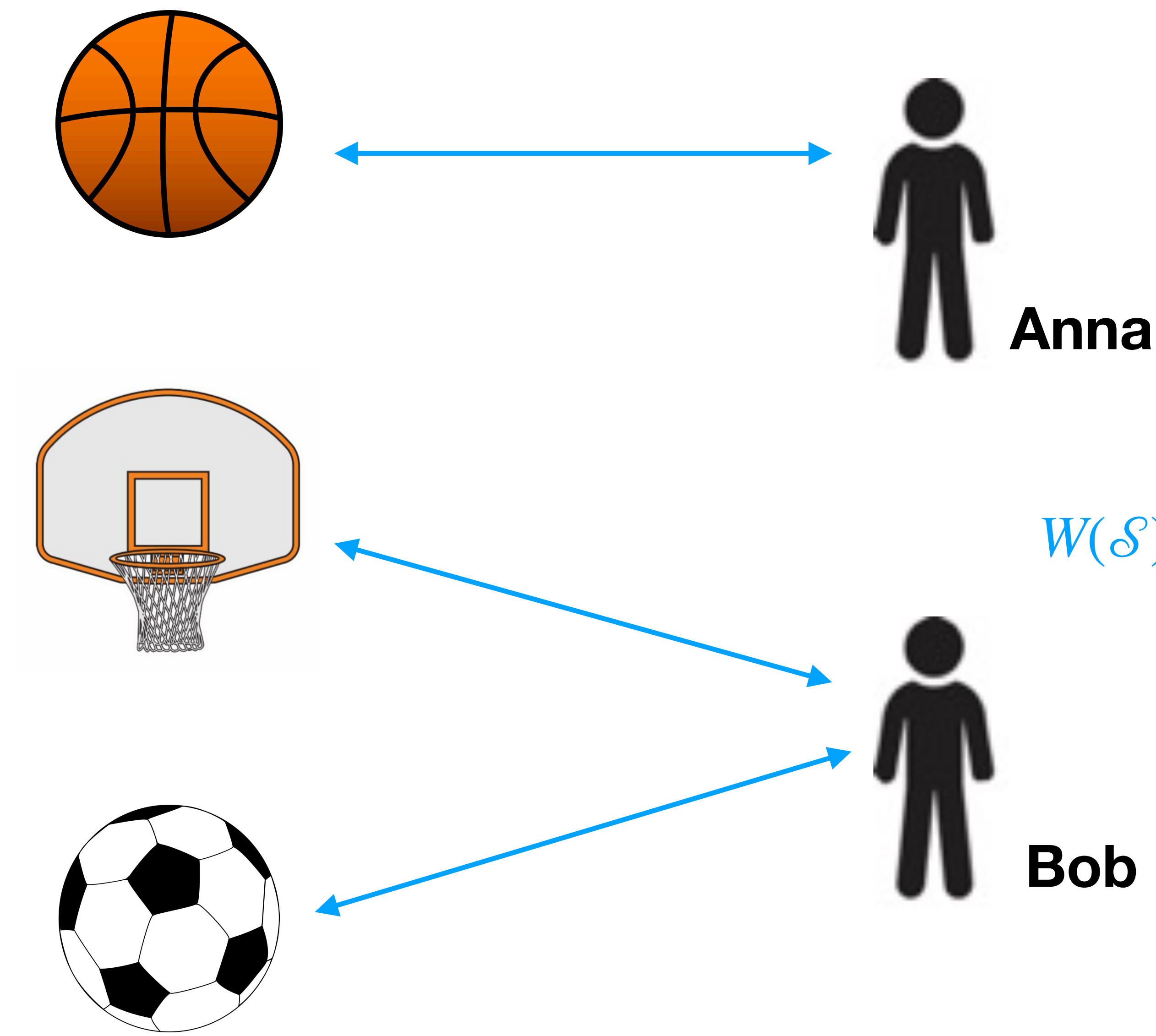
$S_i \cap S_j = \varnothing$

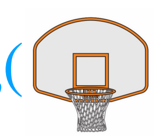
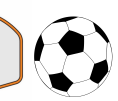
Image credits: http://clipart-library.com/

Model

Set of indivisible **goods**, G

Set of **buyers**, N



$W(\mathcal{S}) = v_{anna}($  $) + v_{bob}($   $)$

$= 1 + 1$

$$\mathcal{S}^* \in \arg\max_{\mathcal{S}} W(\mathcal{S})$$

Allocation, $\mathcal{S} = (S_1, \ldots, S_n)$

$S_i \cap S_j = \varnothing$

Image credits: http://clipart-library.com/

Model

Set of indivisible **goods**, G

Set of **buyers**, N

Anna

Bob

Image credits: http://clipart-library.com/

Model

Set of indivisible **goods**, G

Set of **buyers**, N

Prices per good

p_1

p_2

p_3

Anna

Bob

Image credits: http://clipart-library.com/

Model

Set of indivisible **goods**, G

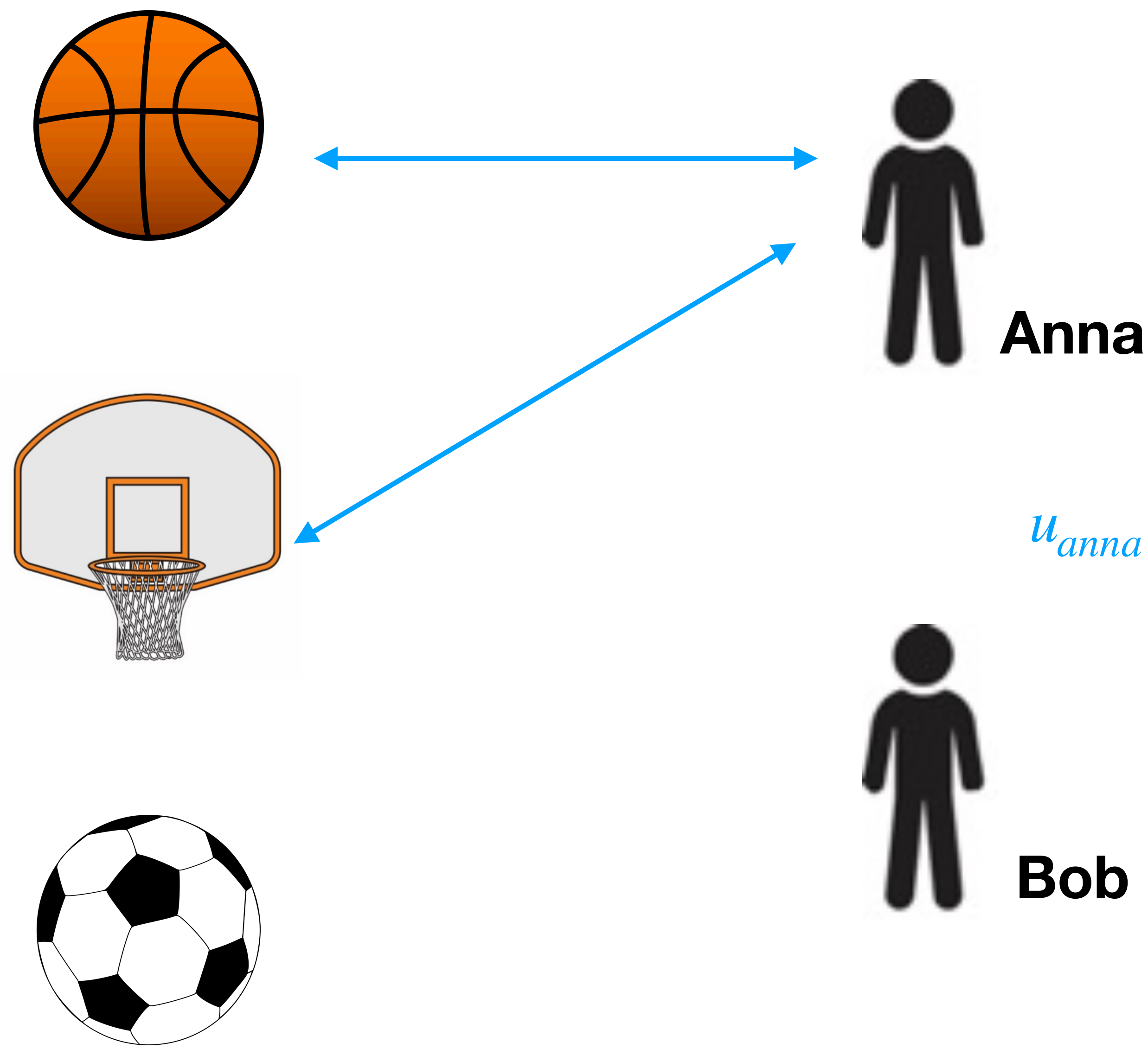
Set of **buyers**, N

Prices per good

p_1

p_2

p_3



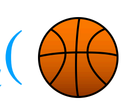
$u_{anna} = v_{anna}($   $) - (p_1 + p_2)$

Image credits: http://clipart-library.com/

Model

Set of indivisible **goods**, G

Set of **buyers**, N

Prices per good

p_1

p_2

p_3

Anna

Outcome: $(\mathcal{S}, \overrightarrow{p})$

Bob

Image credits: <u>http://clipart-library.com/</u>

Combinatorial Markets, Examples

Combinatorial Markets, Examples

- Allocation of landing and take-off slots at airports

Combinatorial Markets, Examples

- Allocation of landing and take-off slots at airports

- Placement of internet advertisement

Combinatorial Markets, Examples

- Allocation of landing and take-off slots at airports

- Placement of internet advertisement

- Spectrum auctions, (2014 Canadian 700 MHZ ~$5 billion)

Combinatorial Markets, Examples

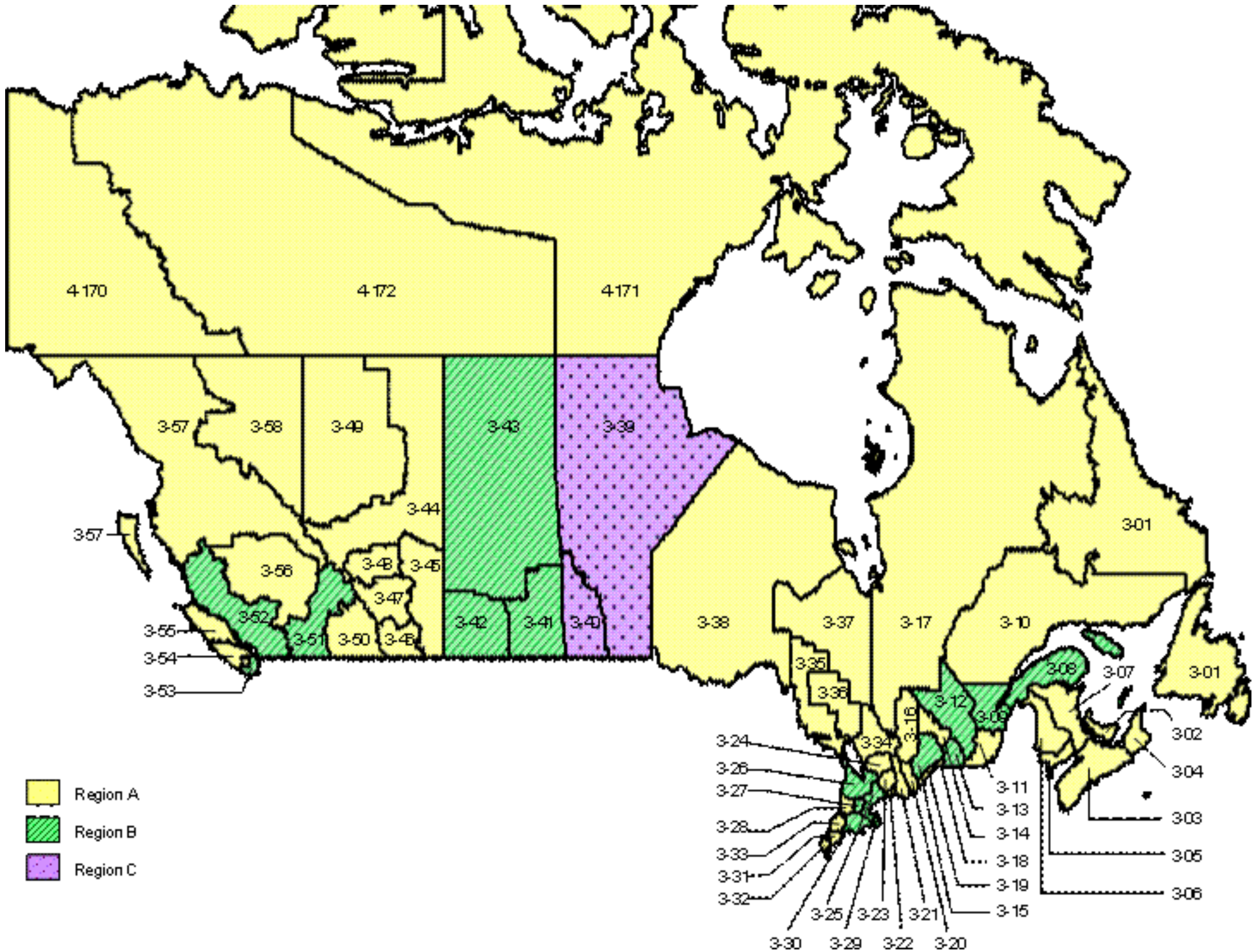
- Allocation of landing and take-off slots at airports

- Placement of internet advertisement

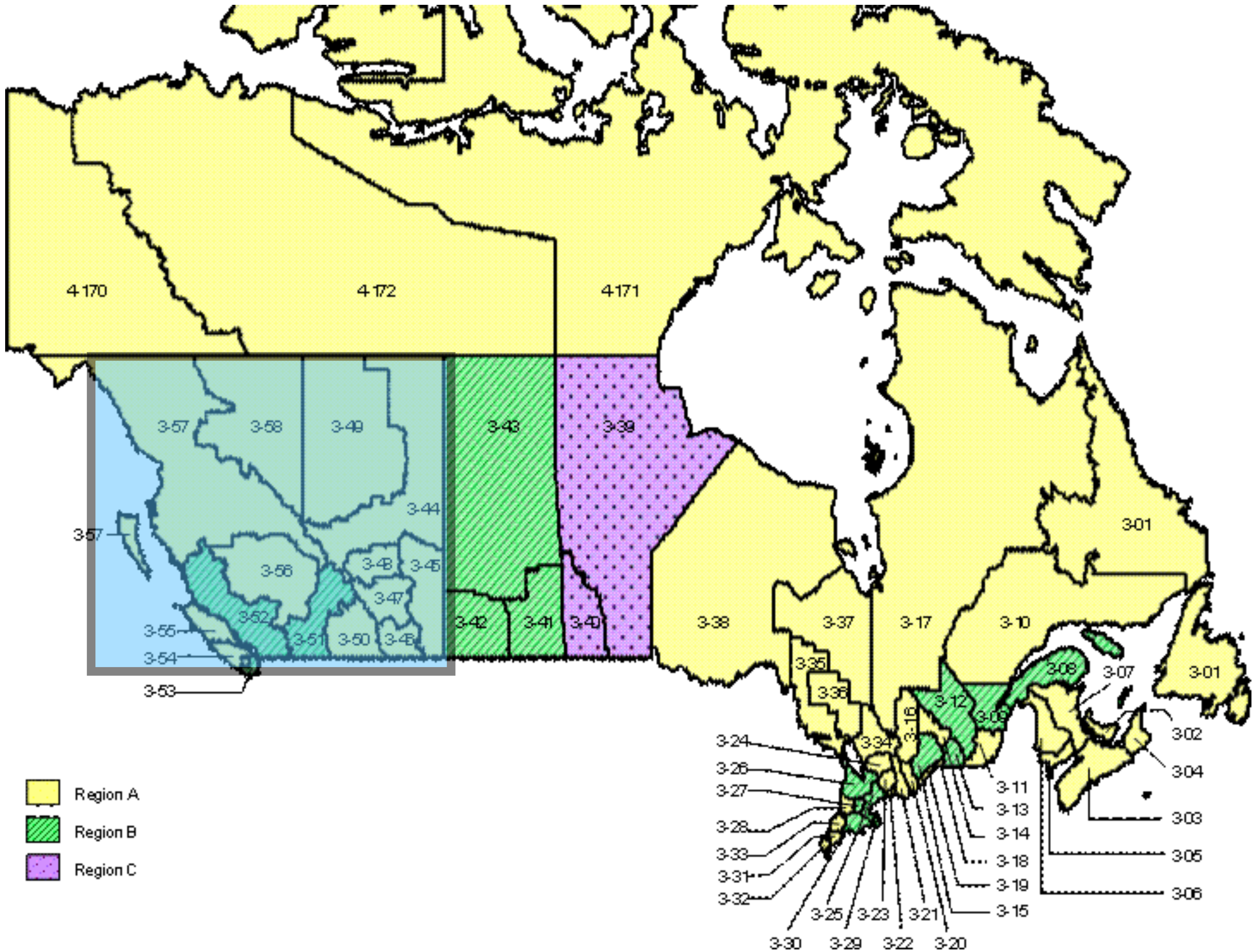
- Spectrum auctions, (2014 Canadian 700 MHZ ~$5 billion)

etc

Example - Electromagnetic Spectrum Allocation

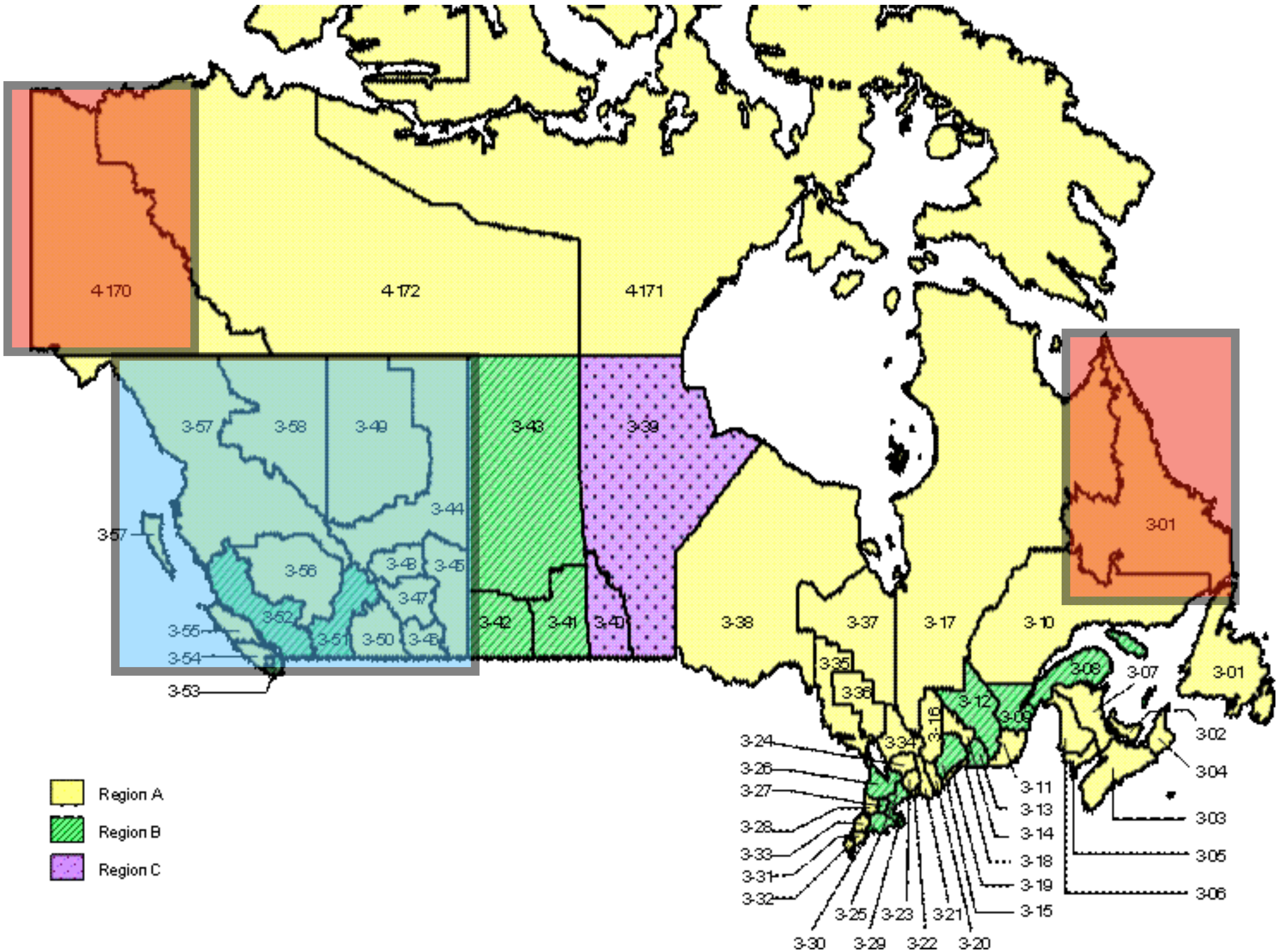


Example - Electromagnetic Spectrum Allocation



v_i(**blue region**)

Example - Electromagnetic Spectrum Allocation



v_i(**blue region**) > v_i(**red region**)

Model, Competitive Equilibrium

- An outcome $(\mathcal{S}, \overrightarrow{p})$ is a competitive equilibrium (CE) if

Model, Competitive Equilibrium

- An outcome $(\mathcal{S}, \overrightarrow{p})$ is a competitive equilibrium (CE) if
  - All buyers are **happy**: $S_i \in \arg\max_{S \subseteq G} v_i(S) - \sum_{j \in G} p_j$

Model, Competitive Equilibrium

- An outcome $(\mathcal{S}, \overrightarrow{p})$ is a competitive equilibrium (CE) if
  - All buyers are **happy**: $S_i \in \arg\max_{S \subseteq G} v_i(S) - \sum_{j \in G} p_j$
  - At prices $\overrightarrow{p}$, the seller maximizes its **revenue** over all allocations

Model, Competitive Equilibrium

- An outcome $(\mathcal{S}, \overrightarrow{p})$ is a ε- competitive equilibrium (CE) if
  - All buyers are **happy**: $S_i \in \arg\max_{S \subseteq G} v_i(S) - \sum_{j \in G} p_j - \varepsilon$
  - At prices $\overrightarrow{p}$, the seller maximizes its **revenue** over all allocations

Outline - Combinatorial Markets

- Model and Examples
- Noisy Combinatorial Markets
- Revisiting Pruning and Experiments

Outline - Combinatorial Markets

- ~~Model and Examples~~
- Noisy Combinatorial Markets
- Revisiting Pruning and Experiments

Noisy Combinatorial Markets

- Assumption: buyers **exactly** know their **values** for all bundles

Noisy Combinatorial Markets

- Assumption: buyers **exactly** know their **values** for all bundles
- However, this may not always be the case. Why?

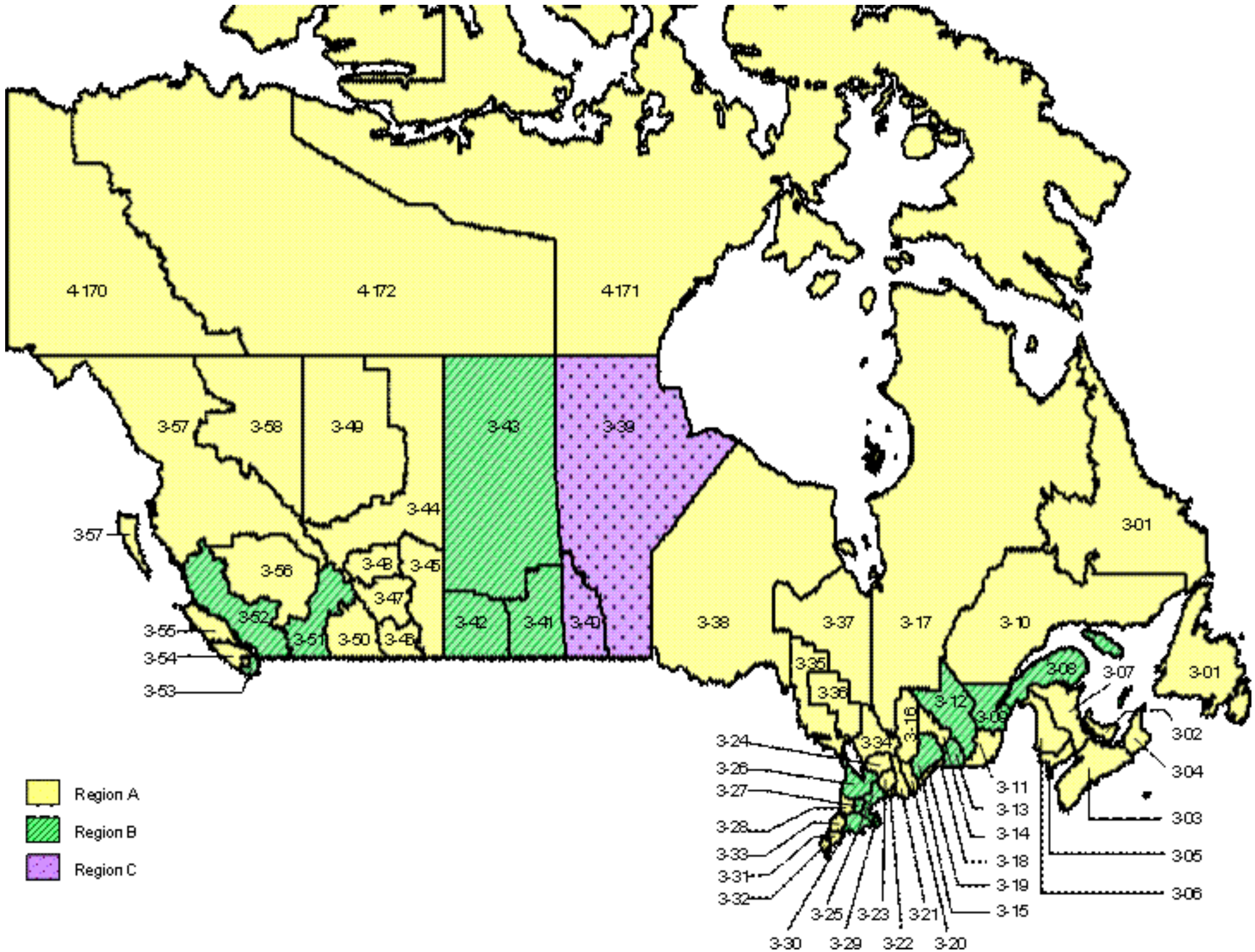
Noisy Combinatorial Markets

- Assumption: buyers **exactly** know their **values** for all bundles
- However, this may not always be the case. Why?
  - Value for bundle might depend on unobservable factors, e.g., whether an event occurs or not

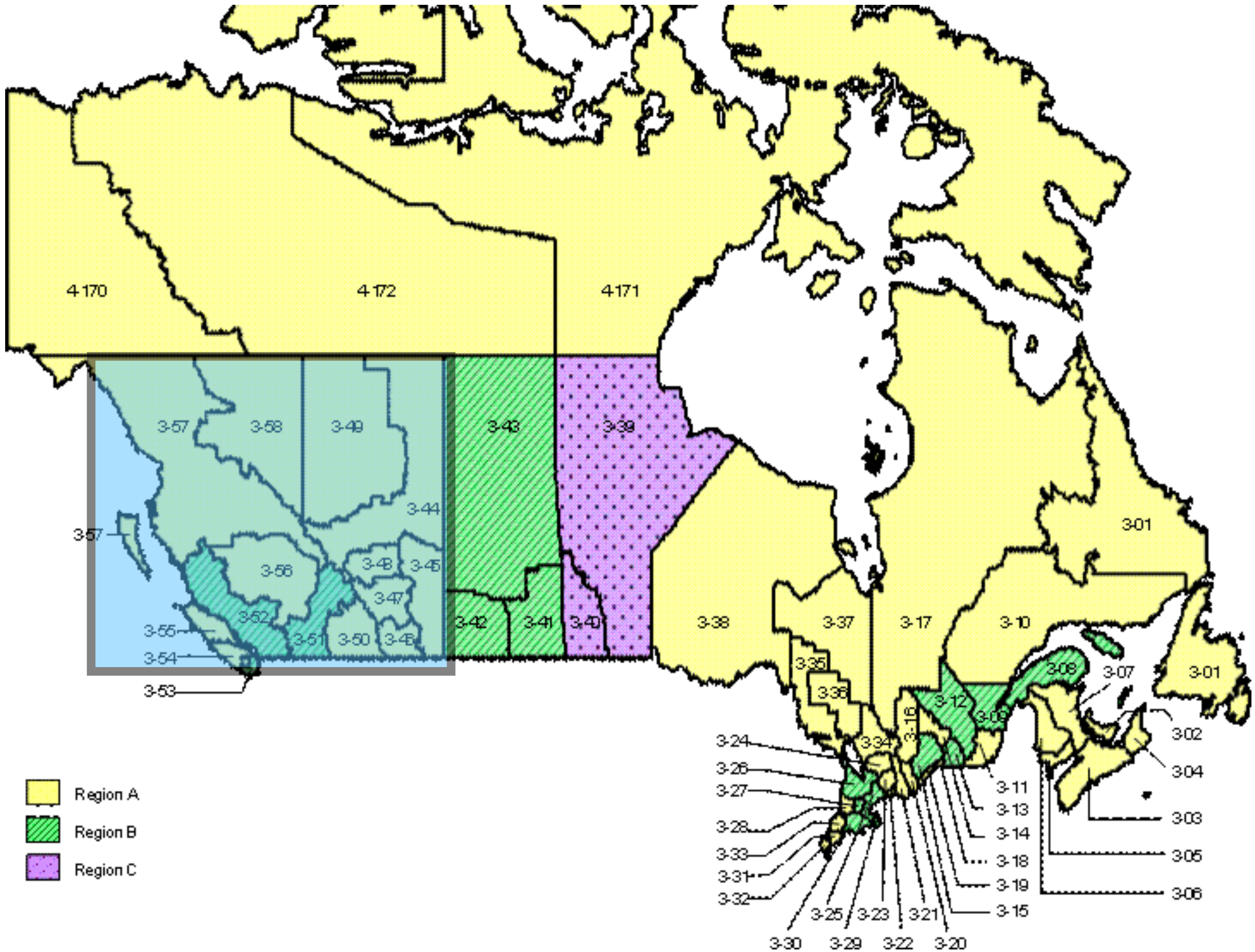
Noisy Combinatorial Markets

- Assumption: buyers **exactly** know their **values** for all bundles
- However, this may not always be the case. Why?
  - Value for bundle might depend on unobservable factors, e.g., whether an event occurs or not
  - There might be too many goods, so heuristic or approximate methods might be used to obtain value estimates

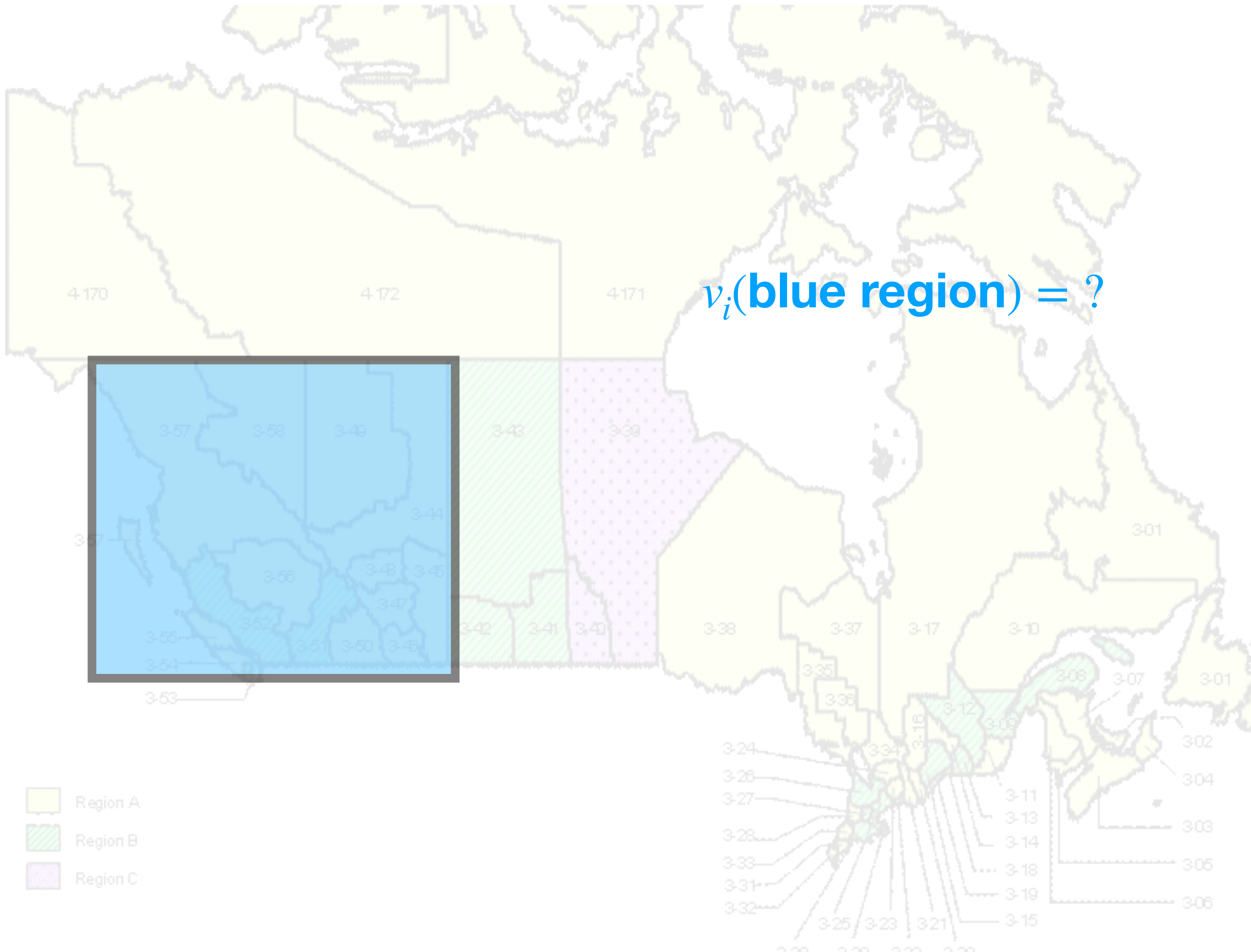
Noisy Combinatorial Markets (cont.)



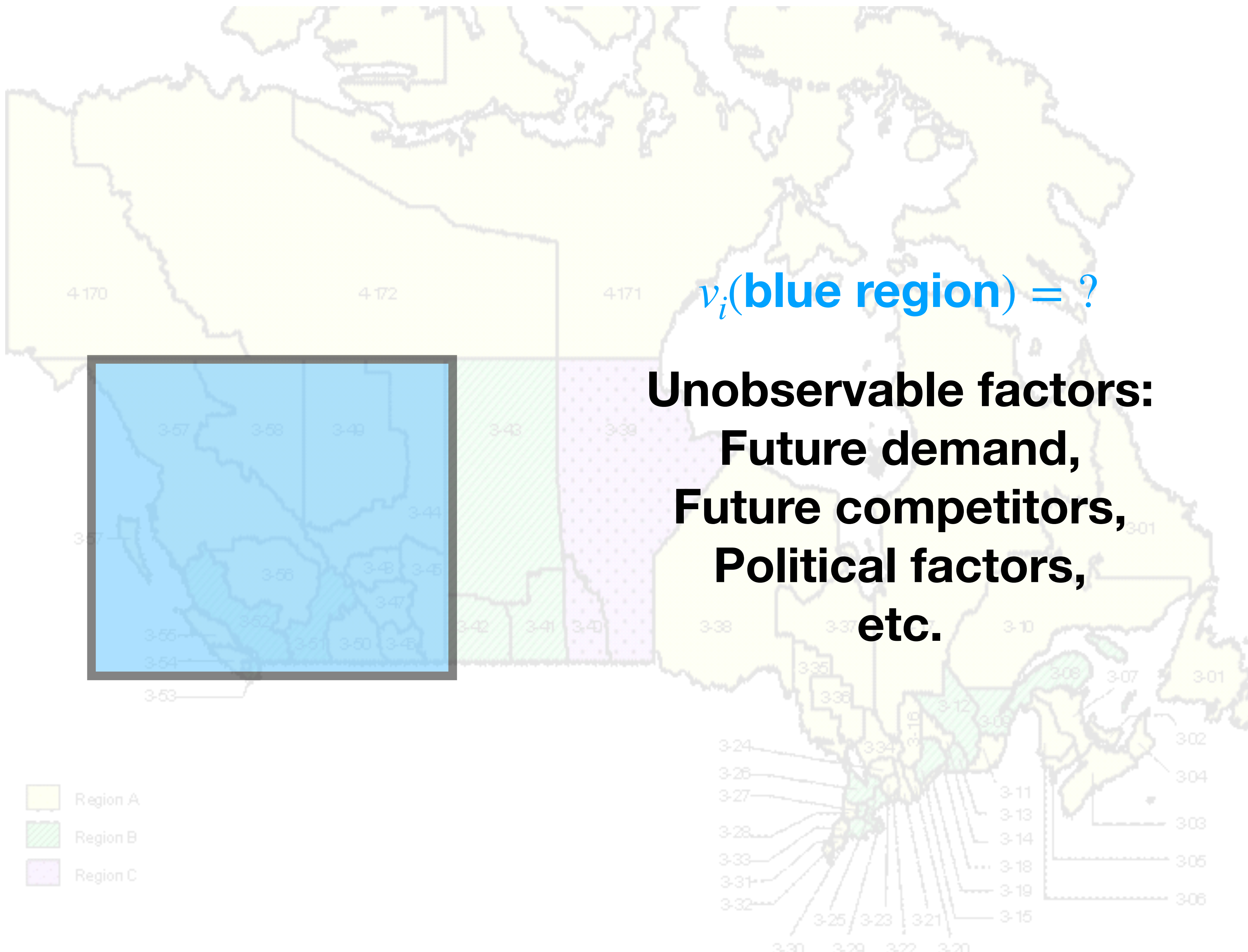
Noisy Combinatorial Markets (cont.)



Noisy Combinatorial Markets (cont.)



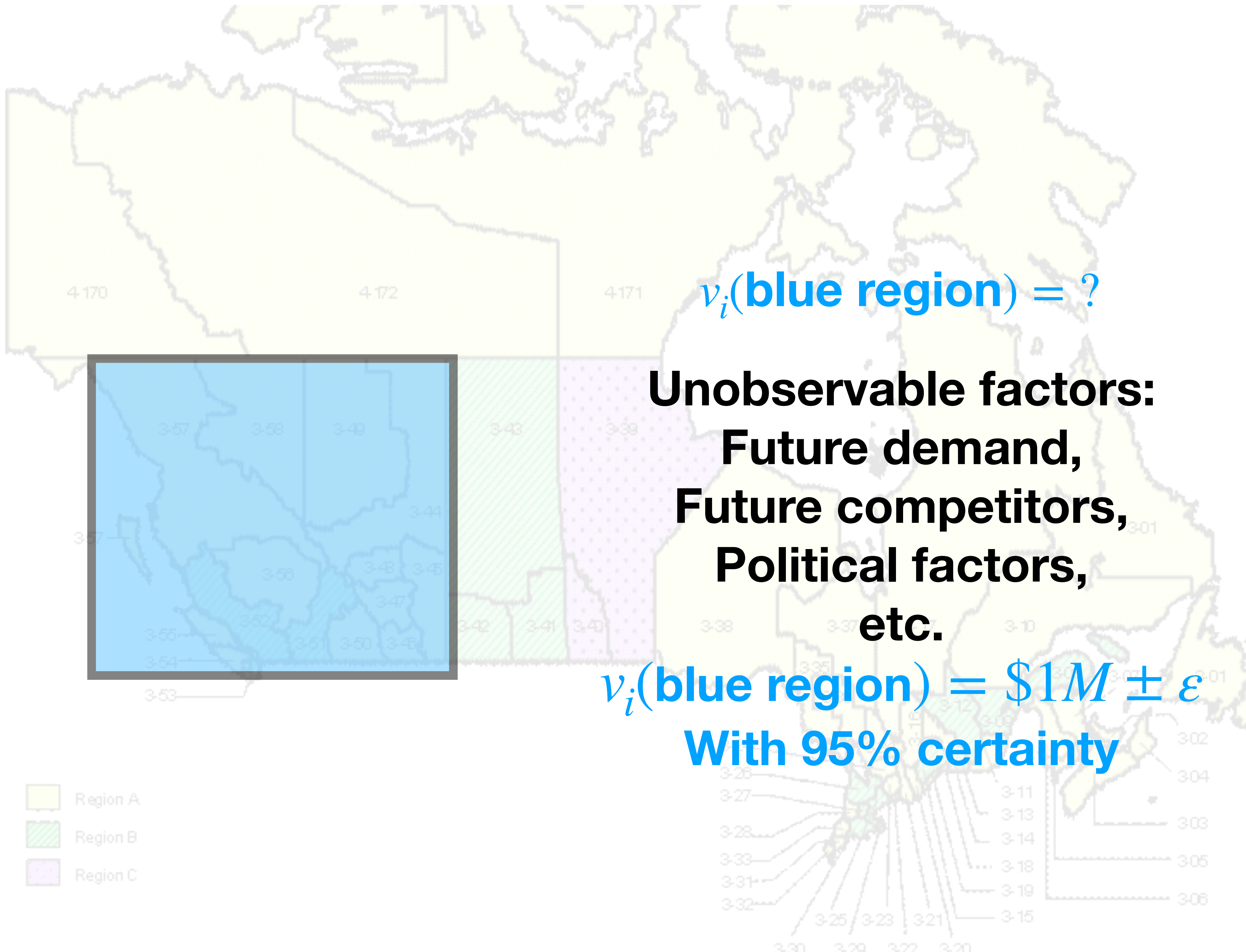
Noisy Combinatorial Markets (cont.)



v_i(**blue region**) $= ?$

Unobservable factors:
Future demand,
Future competitors,
Political factors,
etc.

Noisy Combinatorial Markets (cont.)



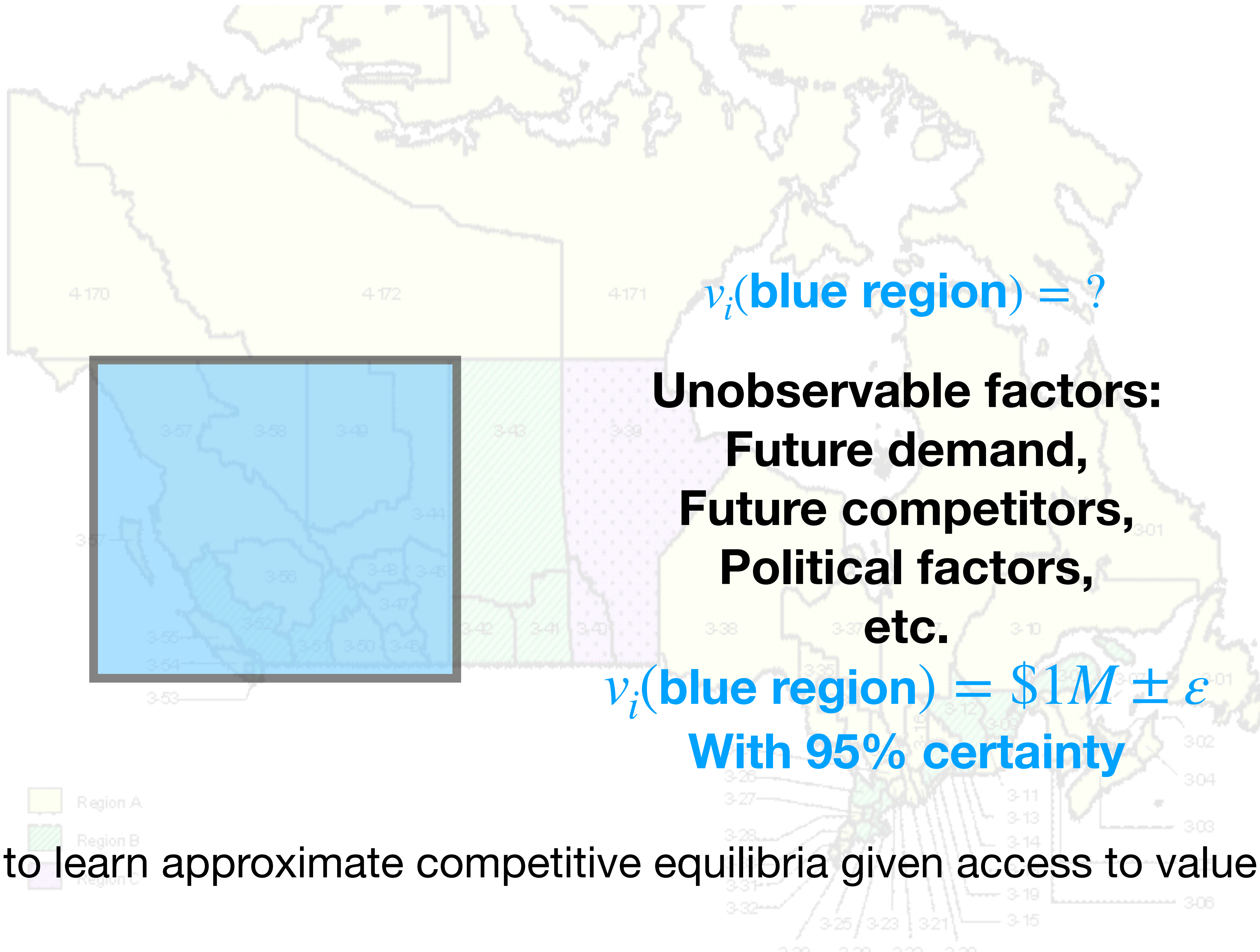
v_i(**blue region**) $= ?$

Unobservable factors:
Future demand,
Future competitors,
Political factors,
etc.

v_i(**blue region**) $= \$1M \pm \varepsilon$

With 95% certainty

Noisy Combinatorial Markets (cont.)



v_i(**blue region**) $= ?$

Unobservable factors:
Future demand,
Future competitors,
Political factors,
etc.

v_i(**blue region**) $= \$1M \pm \varepsilon$
With 95% certainty

Our goal: to learn approximate competitive equilibria given access to value estimates

Learning - déjà vu

- $v_i(S, x)$ is consumer ith's **conditional value** for bundle S.

Learning - déjà vu

- $v_i(S, x)$ is consumer ith's **conditional value** for bundle S.

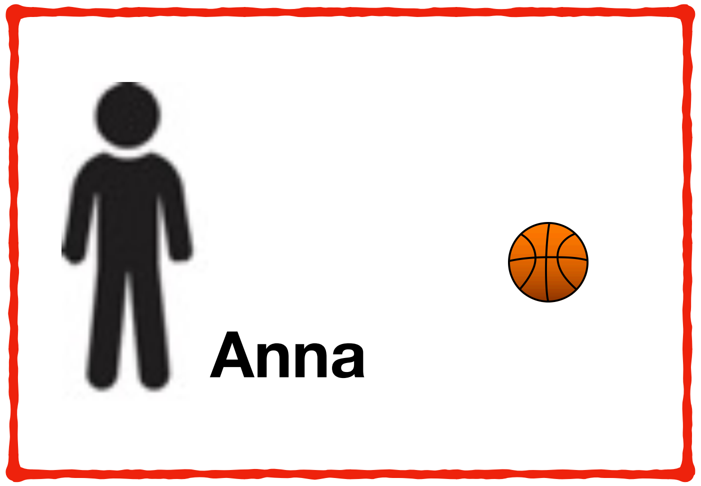
- $\bar{v}_i(S) = \mathbb{E}_{x \sim \mathscr{D}}[v_i(S; x)]$ is consumer ith's **expected value** for bundle S.

Learning - déjà vu

- $v_i(S, x)$ is consumer ith's **conditional value** for bundle S.
- $\bar{v}_i(S) = \mathbb{E}_{x\sim\mathscr{D}}[v_i(S; x)]$ is consumer ith's **expected value** for bundle S.
- Given m samples: $v_i(S; x_1), v_i(S; x_2), \cdots, v_i(S; x_m)$
  consumer ith's **empirical value** is the average: $\hat{v}_i(S) = \frac{1}{m}\sum_{k=1}^{m} v_i(S; x_k)$.

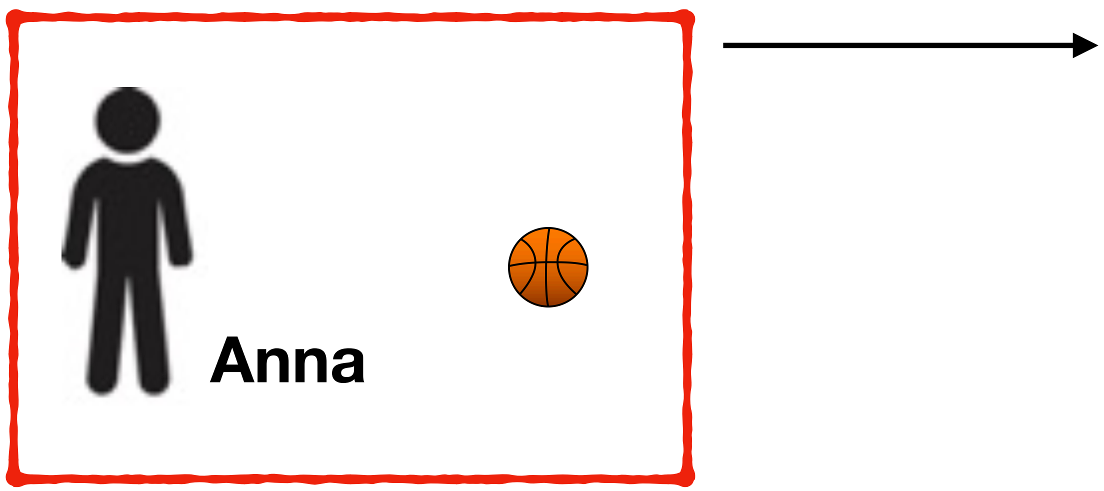
Learning - déjà vu

- $v_i(S, x)$ is consumer ith's **conditional value** for bundle S.
- $\bar{v}_i(S) = \mathbb{E}_{x \sim \mathcal{D}}[v_i(S; x)]$ is consumer ith's **expected value** for bundle S.
- Given m samples: $v_i(S; x_1), v_i(S; x_2), \cdots, v_i(S; x_m)$
  consumer ith's **empirical value** is the average: $\hat{v}_i(S) = \frac{1}{m} \sum_{k=1}^{m} v_i(S; x_k)$.



Learning - déjà vu

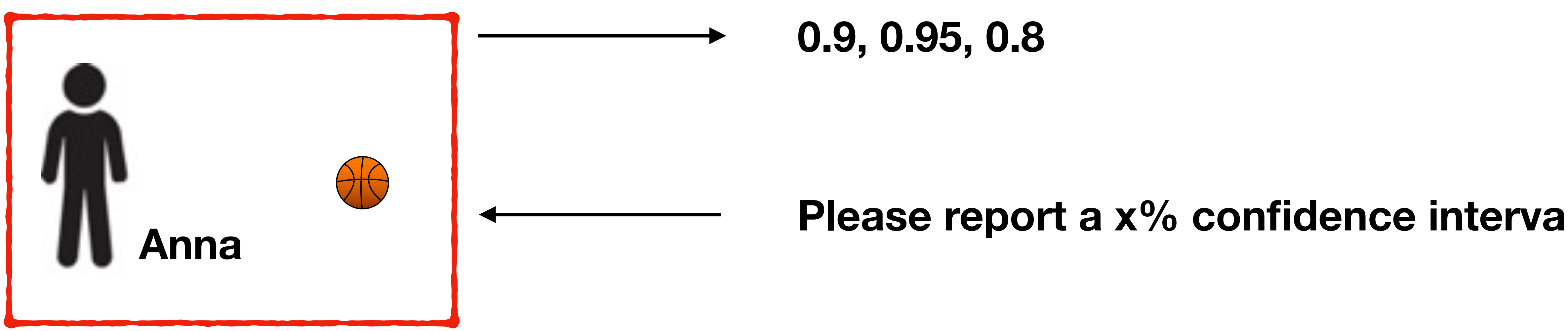
- $v_i(S, x)$ is consumer ith's **conditional value** for bundle S.
- $\bar{v}_i(S) = \mathbb{E}_{x \sim \mathcal{D}}[v_i(S; x)]$ is consumer ith's **expected value** for bundle S.
- Given m samples: $v_i(S; x_1), v_i(S; x_2), \cdots, v_i(S; x_m)$
  consumer ith's **empirical value** is the average: $\hat{v}_i(S) = \frac{1}{m} \sum_{k=1}^{m} v_i(S; x_k)$.



0.9, 0.95, 0.8

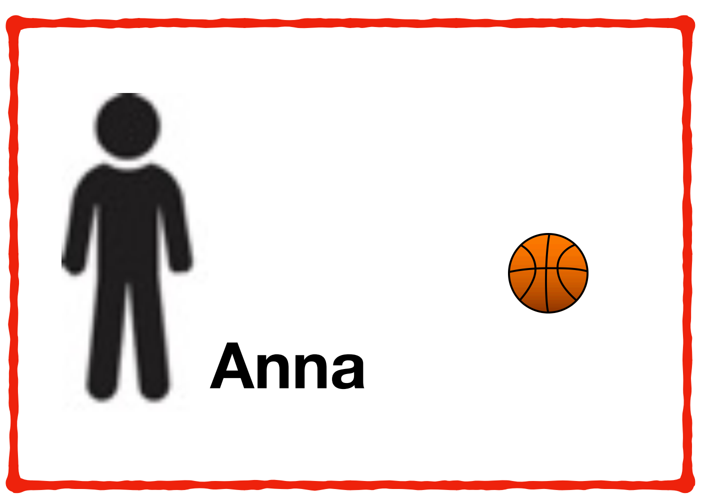
Learning - déjà vu

- $v_i(S, x)$ is consumer ith's **conditional value** for bundle S.
- $\bar{v}_i(S) = \mathbb{E}_{x \sim \mathcal{D}}[v_i(S; x)]$ is consumer ith's **expected value** for bundle S.
- Given m samples: $v_i(S; x_1), v_i(S; x_2), \cdots, v_i(S; x_m)$
  consumer ith's **empirical value** is the average: $\hat{v}_i(S) = \frac{1}{m} \sum_{k=1}^{m} v_i(S; x_k)$.



Learning - déjà vu

- $v_i(S, x)$ is consumer ith's **conditional value** for bundle S.
- $\bar{v}_i(S) = \mathbb{E}_{x \sim \mathcal{D}}[v_i(S; x)]$ is consumer ith's **expected value** for bundle S.
- Given m samples: $v_i(S; x_1), v_i(S; x_2), \cdots, v_i(S; x_m)$
  consumer ith's **empirical value** is the average: $\hat{v}_i(S) = \frac{1}{m}\sum_{k=1}^{m} v_i(S; x_k)$.



→ **0.9, 0.95, 0.8**

← **Please report a x% confidence interval**

→ $1 \pm \epsilon$

Outline - Combinatorial Markets

- ~~Model and Examples~~
- Noisy Combinatorial Markets
- Revisiting Pruning and Experiments

Outline - Combinatorial Markets

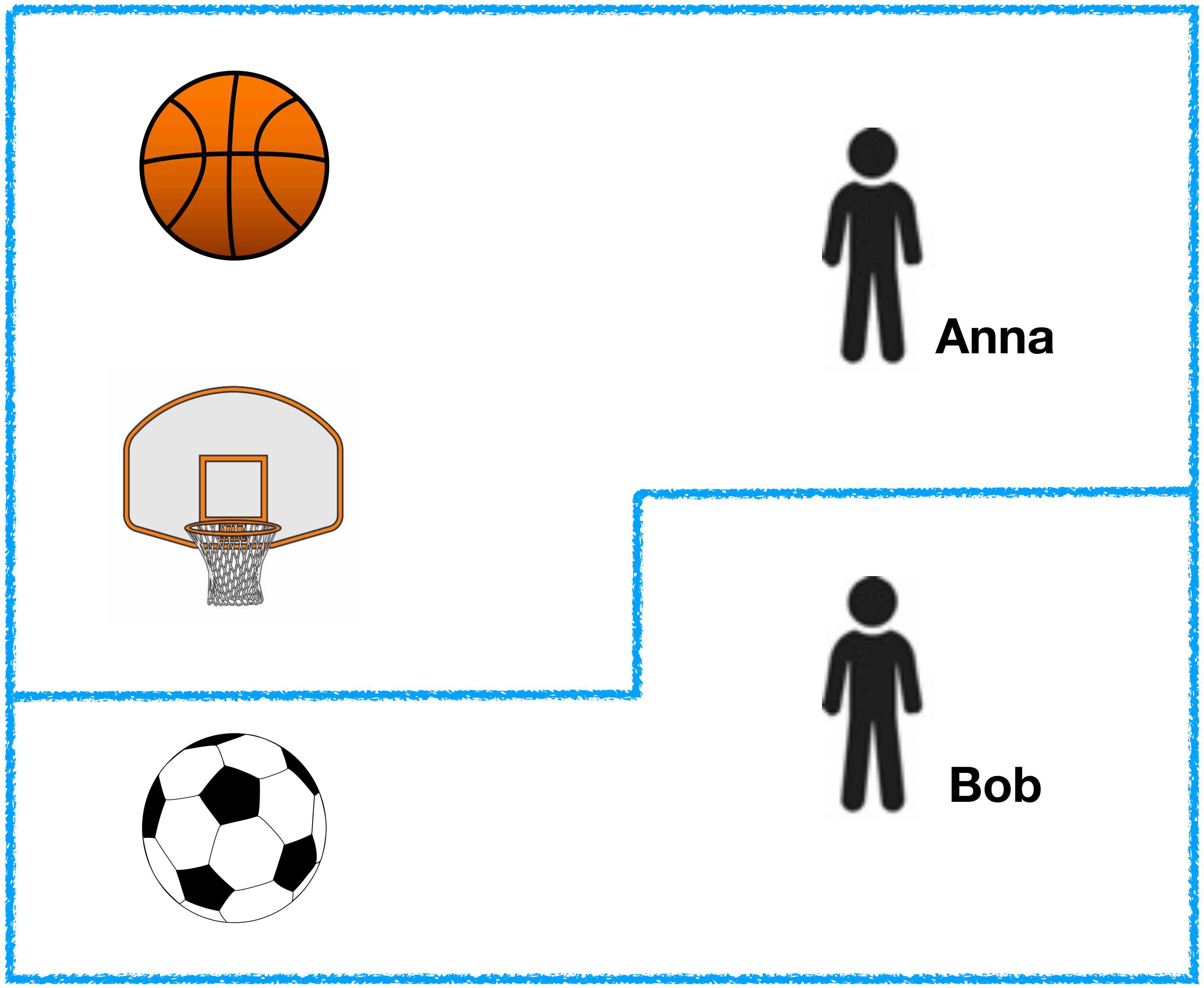
- ~~Model and Examples~~
- ~~Noisy Combinatorial Markets~~
- Revisiting Pruning and Experiments

Revisiting Pruning

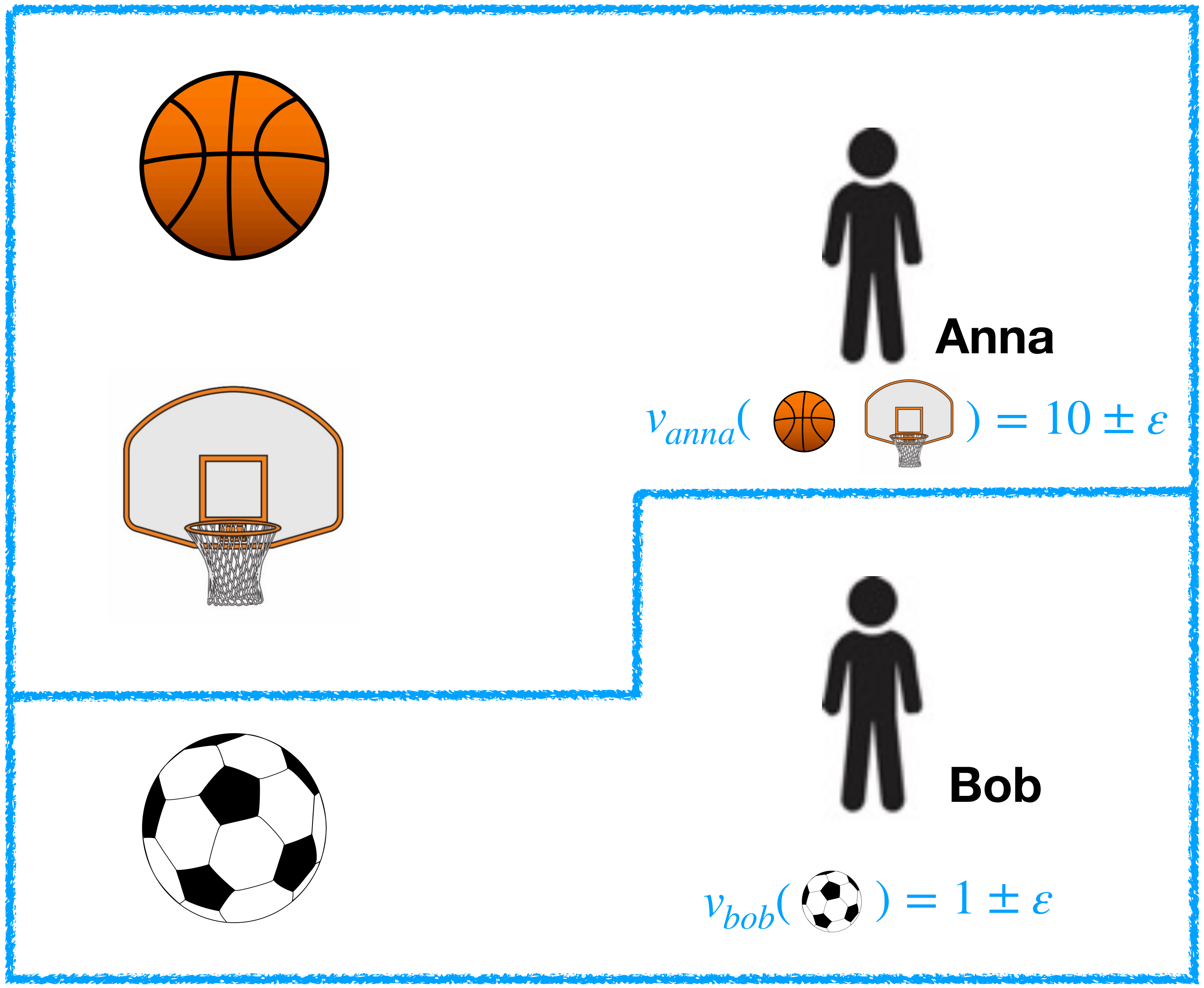
Anna

Bob

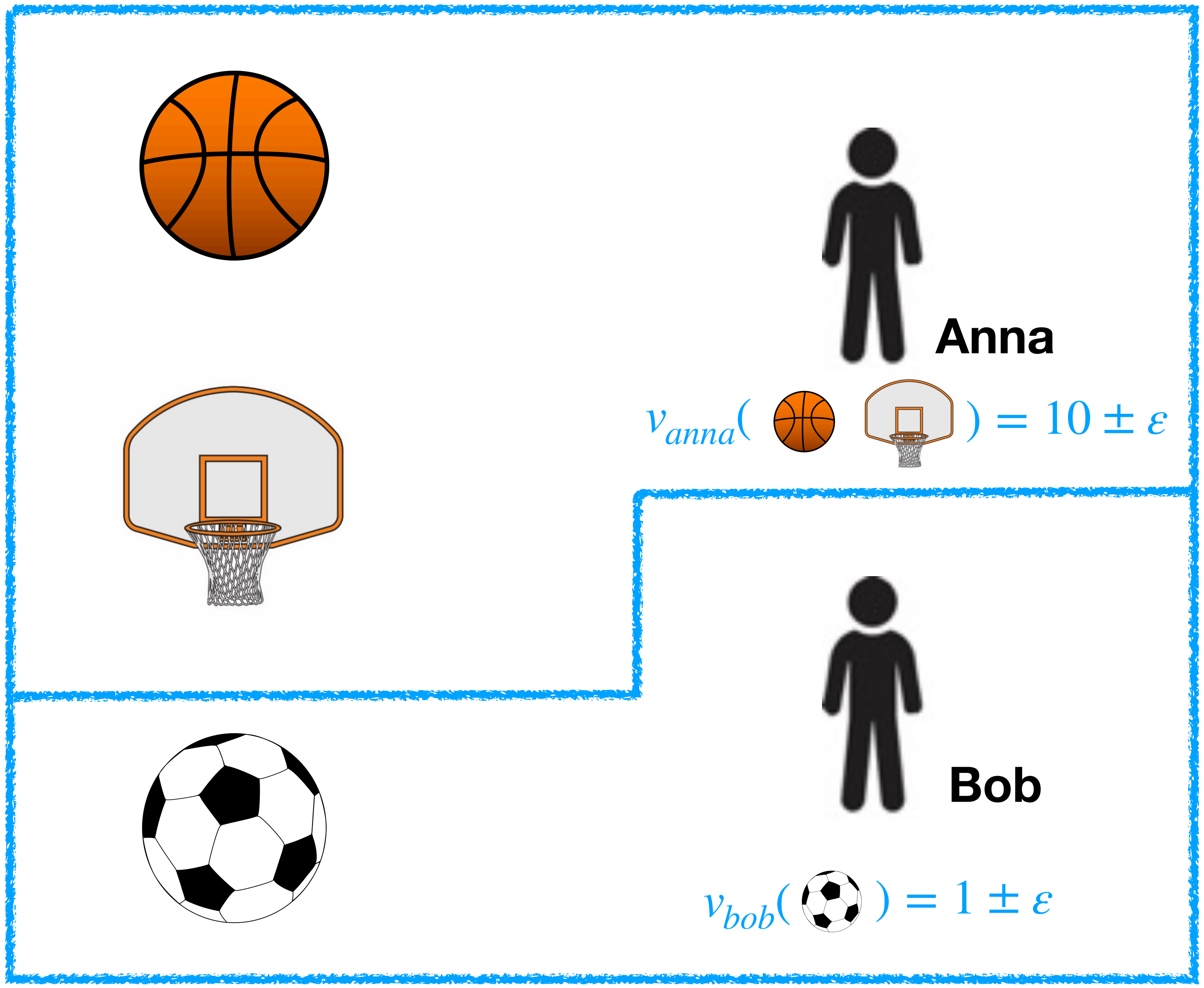
Revisiting Pruning



Revisiting Pruning



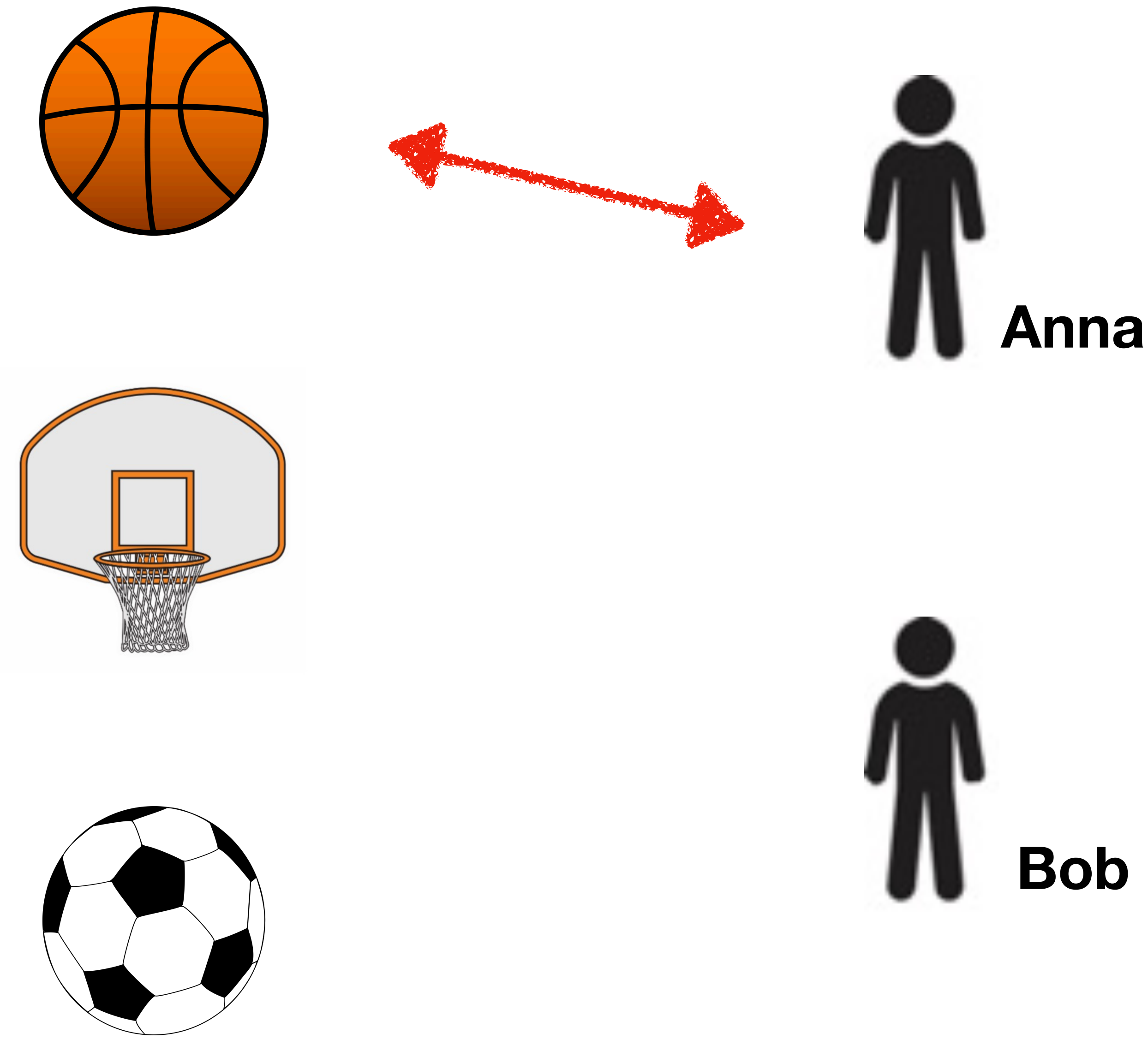
Revisiting Pruning



Blue Allocation's Welfare

$$W^* = (10 \pm \varepsilon) + (1 \pm \varepsilon)$$

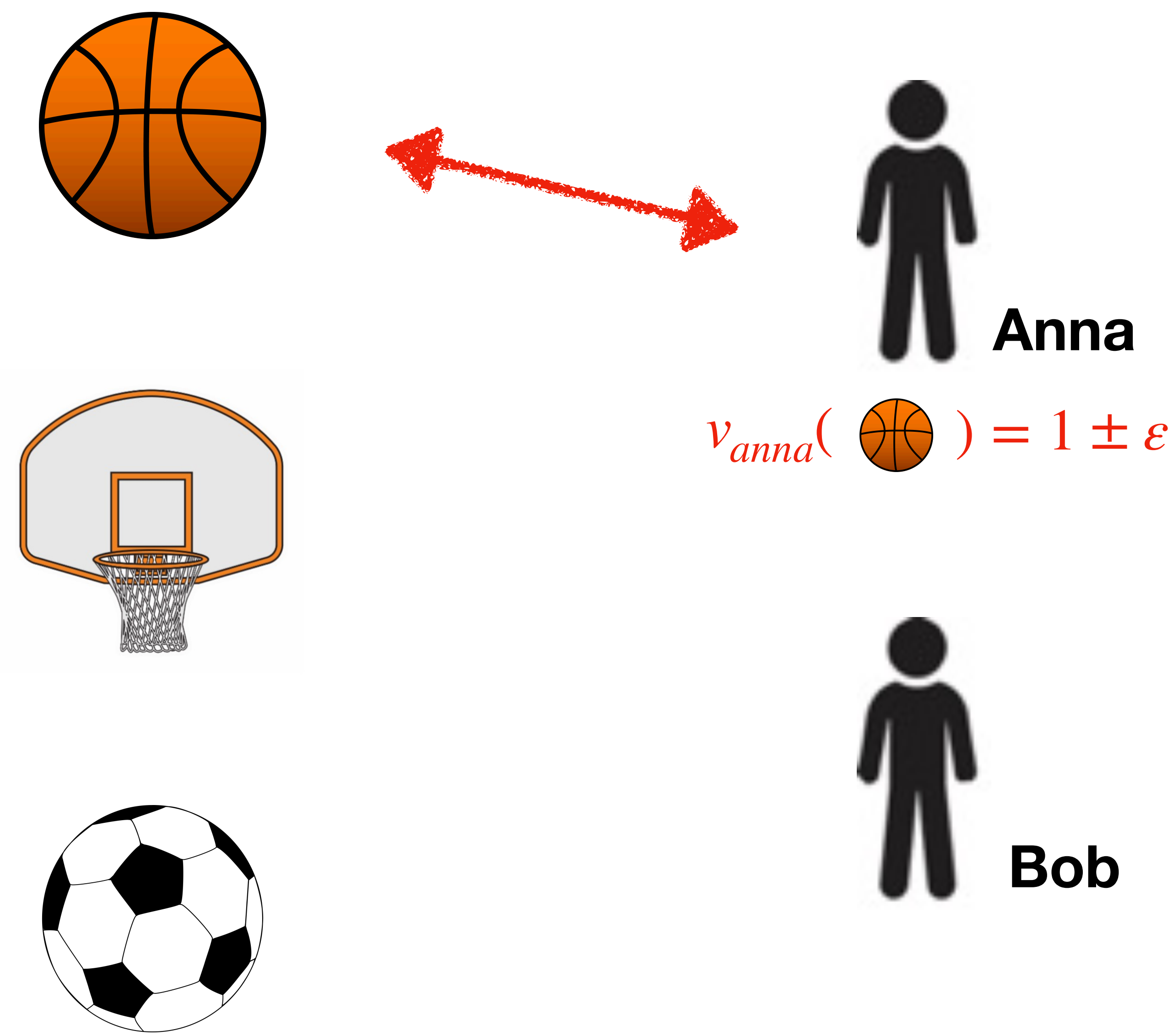
Revisiting Pruning



Blue Allocation's Welfare

$$W^* = (10 \pm \varepsilon) + (1 \pm \varepsilon)$$

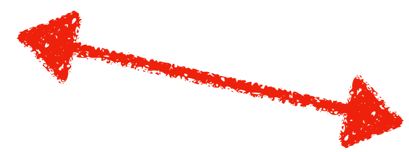
Revisiting Pruning



Blue Allocation's Welfare

$W^* = (10 \pm \varepsilon) + (1 \pm \varepsilon)$

Revisiting Pruning



Anna

$v_{anna}($  $) = 1 \pm \varepsilon$

Bob

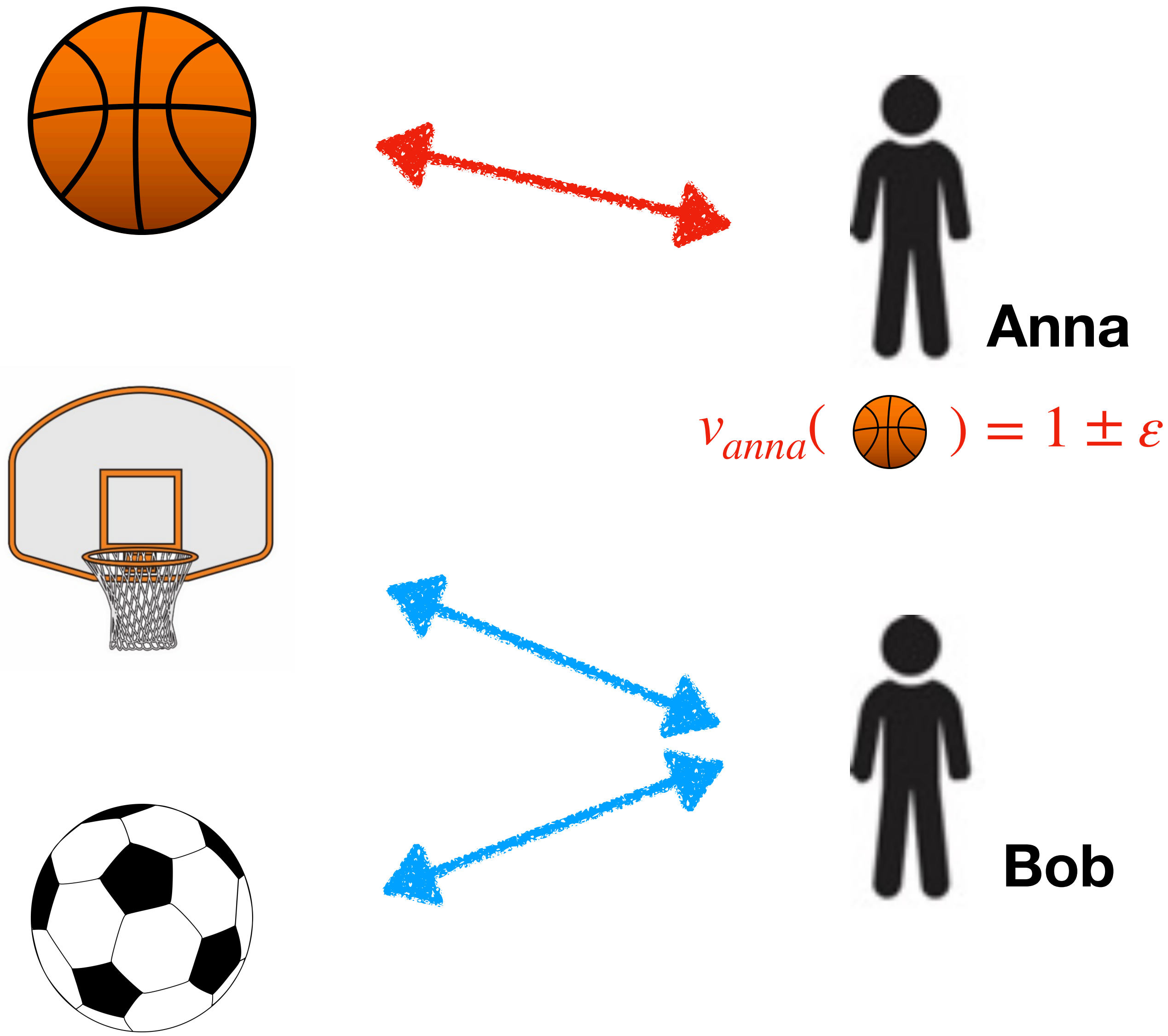
Blue Allocation's Welfare

$W^* = (10 \pm \varepsilon) + (1 \pm \varepsilon)$

 **Anna gets** 

$W_1^* = (1 \pm \varepsilon)$

Revisiting Pruning



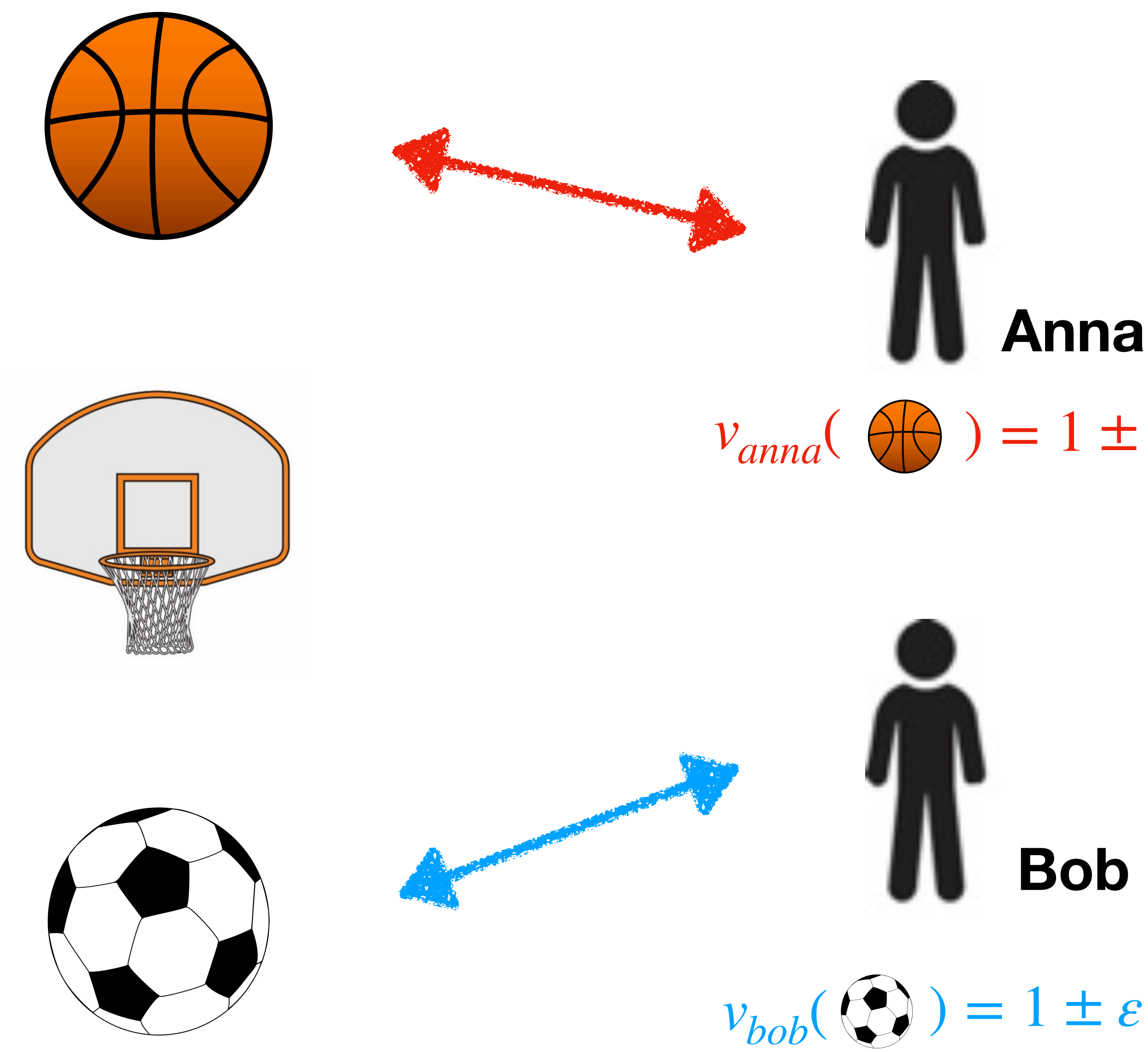
Blue Allocation's Welfare

$$W^* = (10 \pm \varepsilon) + (1 \pm \varepsilon)$$

Anna gets 🏀

$$W_1^* = (1 \pm \varepsilon)$$

Revisiting Pruning



$v_{anna}(\ \text{basketball}\ ) = 1 \pm \varepsilon$

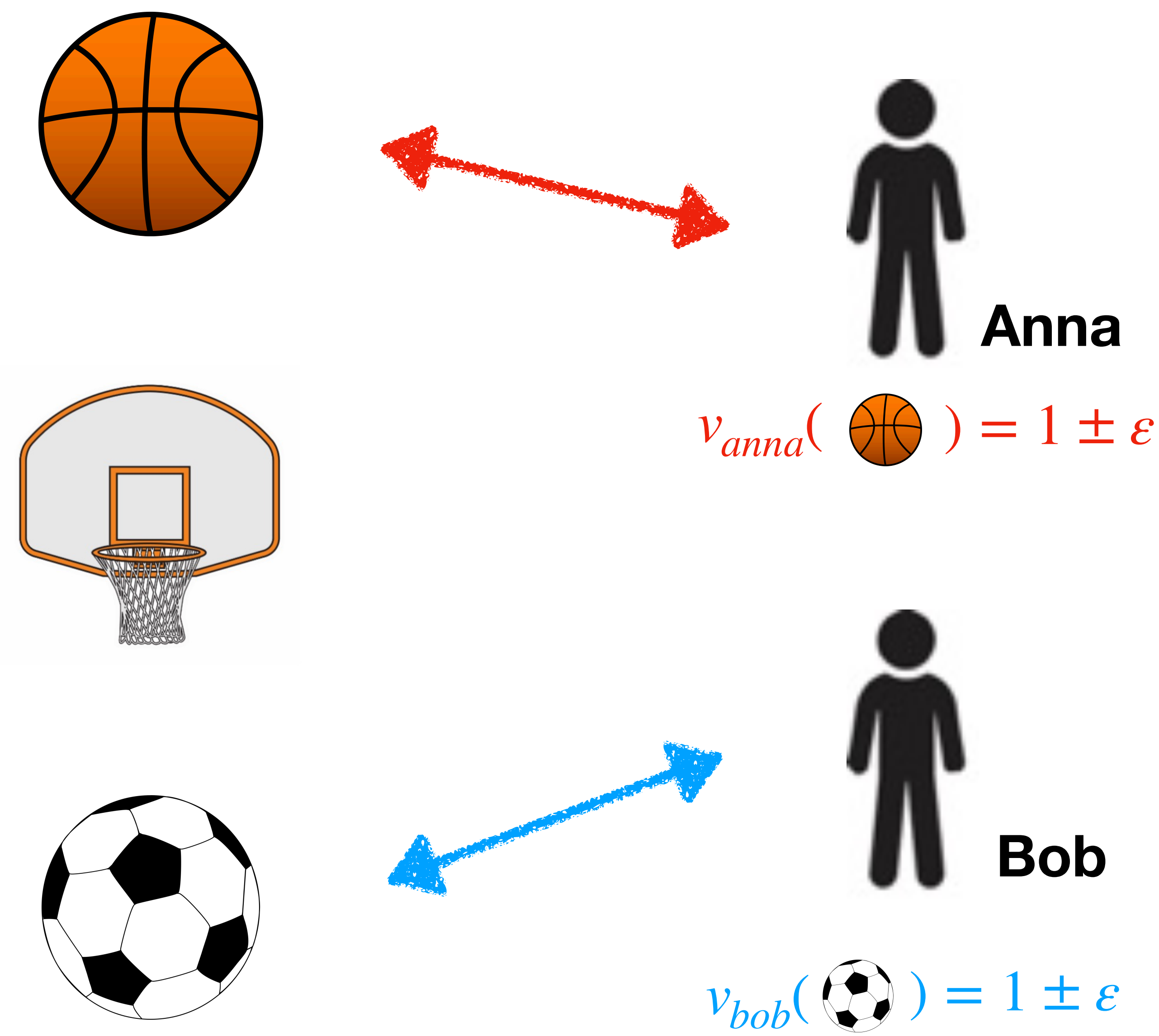
$v_{bob}(\ \text{soccer ball}\ ) = 1 \pm \varepsilon$

Blue Allocation's Welfare

$W^* = (10 \pm \varepsilon) + (1 \pm \varepsilon)$

$W_1^* = (1 \pm \varepsilon)$

Revisiting Pruning



Blue Allocation's Welfare

$$W^* = (10 \pm \varepsilon) + (1 \pm \varepsilon)$$

$$W_1^* = (1 \pm \varepsilon) + (1 \pm \varepsilon)$$

Revisiting Pruning

In the worst case for the blue allocation:

$$W^* = 11 - 2\varepsilon$$

Blue Allocation's Welfare

$$W^* = (10 \pm \varepsilon) + (1 \pm \varepsilon)$$

Anna gets

$$W_1^* = (1 \pm \varepsilon) + (1 \pm \varepsilon)$$

Revisiting Pruning

In the worst case for the blue allocation:

$$W^* = 11 - 2\varepsilon$$

In the best case when Anna gets :

$$W_1^* = 2 + 2\varepsilon$$

Blue Allocation's Welfare

$$W^* = (10 \pm \varepsilon) + (1 \pm \varepsilon)$$

 **Anna gets** 

$$W_1^* = (1 \pm \varepsilon) + (1 \pm \varepsilon)$$

Revisiting Pruning

In the worst case for the blue allocation:

$$W^* = 11 - 2\varepsilon$$

In the best case when Anna gets :

$$W_1^* = 2 + 2\varepsilon$$

Exploiting first-welfare theorem of economics, we prove:

Blue Allocation's Welfare

$$W^* = (10 \pm \varepsilon) + (1 \pm \varepsilon)$$

Anna gets 

$$W_1^* = (1 \pm \varepsilon) + (1 \pm \varepsilon)$$

Revisiting Pruning

In the worst case for the blue allocation:

$$W^* = 11 - 2\varepsilon$$

In the best case when Anna gets :

$$W_1^* = 2 + 2\varepsilon$$

Exploiting first-welfare theorem of economics, we prove:

If ε is small enough ($\varepsilon < 9/4$), there is no way that Anna gets just  at **any** competitive equilibrium.

Blue Allocation's Welfare

$$W^* = (10 \pm \varepsilon) + (1 \pm \varepsilon)$$

 **Anna gets** 

$$W_1^* = (1 \pm \varepsilon) + (1 \pm \varepsilon)$$

Revisiting Pruning

In the worst case for the blue allocation:

$$W^* = 11 - 2\varepsilon$$

In the best case when Anna gets :

$$W_1^* = 2 + 2\varepsilon$$

Exploiting first-welfare theorem of economics, we prove:

If ε is small enough ($\varepsilon < 9/4$), there is no way that Anna gets just  at **any** competitive equilibrium.

Conclusion: we can safely ***stop*** learning Anna's value for  and instead focus learning effort elsewhere.

Blue Allocation's Welfare

$$W^* = (10 \pm \varepsilon) + (1 \pm \varepsilon)$$

 **Anna gets** 

$$W_1^* = (1 \pm \varepsilon) + (1 \pm \varepsilon)$$

Pruning for Combinatorial Markets

- Learn a market up to ε (initially big epsilon, “cheap” to learn)

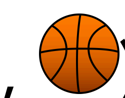
Pruning for Combinatorial Markets

- Learn a market up to ε (initially big epsilon, "cheap" to learn)
  - Solve for welfare-maximizing allocation W^* (blue allocation!)

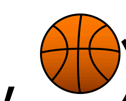
Pruning for Combinatorial Markets

- Learn a market up to ε (initially big epsilon, "cheap" to learn)
  - Solve for welfare-maximizing allocation W^* (blue allocation!)
- For each possible (buyer, bundle) pair, (i, S) (e.g., Anna, )

Pruning for Combinatorial Markets

- Learn a market up to ε (initially big epsilon, “cheap” to learn)
  - Solve for welfare-maximizing allocation W^* (blue allocation!)
- For each possible (buyer, bundle) pair, (i, S) (e.g., Anna, )
  - Assume i gets S

Pruning for Combinatorial Markets

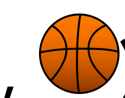
- Learn a market up to ε (initially big epsilon, “cheap” to learn)
  - Solve for welfare-maximizing allocation W^* (blue allocation!)
- For each possible (buyer, bundle) pair, (i, S) (e.g., Anna, )
  - Assume i gets S
  - Solve welfare-max. allocation $W^*_{-(i,S)}$ for remaining buyers, items

Pruning for Combinatorial Markets

- Learn a market up to ε (initially big epsilon, "cheap" to learn)
  - Solve for welfare-maximizing allocation W^* (blue allocation!)
- For each possible (buyer, bundle) pair, (i, S) (e.g., Anna, 🏀)
  - Assume i gets S
  - Solve welfare-max. allocation $W^*_{-(i,S)}$ for remaining buyers, items
  - If $W^* > v_i(S) + W^*_{-(i,S)} + f(\epsilon)$, then ***stop learning*** value $v_i(S)$

Pruning for Combinatorial Markets

Hard computational problem

- Learn a market up to ε (initially big epsilon, "cheap" to learn)
  - Solve for welfare-maximizing allocation W^* (blue allocation!)
- For each possible (buyer, bundle) pair, (i, S) (e.g., Anna, )
  - Assume i gets S

    Hard computational problem

  - Solve welfare-max. allocation $W^*_{-(i,S)}$ for remaining buyers, items
  - If $W^* > v_i(S) + W^*_{-(i,S)} + f(\epsilon)$, then ***stop learning*** value $v_i(S)$

Pruning for Combinatorial Markets

Hard computational problem

- Learn a market up to ε (initially big epsilon, "cheap" to learn)
  - Solve for welfare-maximizing allocation W^* (blue allocation!)
- For each possible (buyer, bundle) pair, (i, S) (e.g., Anna, )
  - Assume i gets S
  - Solve welfare-max. allocation $W^*_{-(i,S)}$ for remaining buyers, items
  - If $W^* > v_i(S) + W^*_{-(i,S)} + f(\epsilon)$, then ***stop learning*** value $v_i(S)$

Hard computational problem

We show it is enough to use an upper bound to retain guarantees

Experimental Setup - Local-Synergy Value Model (LSVM)

T. Scheffel et al.

Table 1 Local-SVM with the preferred items Q and K of two regional bidder. All their positive valued items are *shaded*

A	B	C	D	E	F
G	H	I	J	K	L
M	N	O	P	Q*	R

A	B	C	D	E	F
G	H	I	J	K*	L
M	N	O	P	Q	R

Experimental Setup - Local-Synergy Value Model (LSVM)

T. Scheffel et al.

Table 1 Local-SVM with the preferred items Q and K of two regional bidder. All their positive valued items are *shaded*

A	B	C	D	E	F
G	H	I	J	K	L
M	N	O	P	Q*	R

A	B	C	D	E	F
G	H	I	J	K*	L
M	N	O	P	Q	R

- Five regional bidders and one national bidder

Experimental Setup - Local-Synergy Value Model (LSVM)

T. Scheffel et al.

Table 1 Local-SVM with the preferred items Q and K of two regional bidder. All their positive valued items are *shaded*

A	B	C	D	E	F
G	H	I	J	K	L
M	N	O	P	Q*	R

A	B	C	D	E	F
G	H	I	J	K*	L
M	N	O	P	Q	R

- Five regional bidders and one national bidder
- Large Markets! National bidder alone has value for 2^{18} bundles

Experimental Results - Local-Synergy Value Model (LSVM)

Target Error	% Savings with Pruning (±4%)	Error guarantee (±0.01)	UM Loss (±0.0005)
1.25	18%	0.89	0.0011
2.50	11%	1.78	0.0018
5.00	-7%	3.59	0.0037
10.0	-35%	7.27	0.0072

95% confidence intervals over 50 draws of LSVM markets

Outline - Combinatorial Markets

- ~~Model and Examples~~
- ~~Noisy Combinatorial Markets~~
- Revisiting Pruning and Experiments

Outline - Combinatorial Markets

- ~~Model and Examples~~
- ~~Noisy Combinatorial Markets~~
- ~~Revisiting Pruning and Experiments~~

Summary Part 2

- Extension of simulation-based games methodology to markets
- Development of pruning criteria exploiting economic theory
- Pruning results in substantial sample savings

Part 3:
Empirical Mechanism Design

Empirical Mechanism Design: Designing Mechanisms from Data.
Enrique Areyan Viqueira, Cyrus Cousins, Yasser Mohammad, Amy Greenwald.
Uncertainty in Artificial Intelligence (UAI19).

On Approximate Welfare-and Revenue-Maximizing Equilibria for Size-Interchangeable Bidders.
Enrique Areyan Viqueira, Amy Greenwald, Victor Naroditskiy.
16th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS17).

The "Design" Plan (a.k.a. Outline Part 3)

- Empirical Mechanism Design
- Experiments - Ad Auctions

The "Design" Plan (a.k.a. Outline Part 3)

- Empirical Mechanism Design
- Experiments - Ad Auctions

Mechanism Design

Mechanism design: designing games so that the ensuing behavior of agents, at equilibrium, leads to desirable outcomes.

Image credits: http://clipart-library.com/

Mechanism Design

Mechanism design: designing games so that the ensuing behavior of agents, at equilibrium, leads to desirable outcomes.

Examples abound:

Image credits: http://clipart-library.com/

Mechanism Design

Mechanism design: designing games so that the ensuing behavior of agents, at equilibrium, leads to desirable outcomes.

Examples abound:

- Design of auctions 

Image credits: http://clipart-library.com/

Mechanism Design

Mechanism design: designing games so that the ensuing behavior of agents, at equilibrium, leads to desirable outcomes.

Examples abound:

- Design of auctions

- Designing negotiation protocols

Image credits: http://clipart-library.com/

Mechanism Design

Mechanism design: designing games so that the ensuing behavior of agents, at equilibrium, leads to desirable outcomes.

Examples abound:

- Design of auctions

- Designing negotiation protocols

- Design of college admission systems

Image credits: http://clipart-library.com/

Mechanism Design

Mechanism design: designing games so that the ensuing behavior of agents, at equilibrium, leads to desirable outcomes.

Examples abound:

- Design of auctions

- Designing negotiation protocols

- Design of college admission systems

- etc.

Image credits: http://clipart-library.com/

The Rules of the Game Matter

Bangladesh

Bangladesh raises USD1.7bn from LTE frequency tender

15 Feb 2018

Bangladesh

The Bangladeshi government has raised a total of BDT52.89 billion (USD1.68 billion) from its 4G spectrum auction, far below the expected BDT110 billion figure, with less than 30% of the 46.4MHz of spectrum put up for sale bought in the tender, The Daily Star writes. Shahjahan Mahmood, chairman of the BTRC, said the regulator was 'not happy' with the results of the auction, adding that the operators will have another opportunity to acquire spectrum at the same price within the next six months.

Market leader GrameenPhone will pay USD408 billion for 5MHz in the 1800MHz band, in addition to a fee to convert its current holdings in the 900MHz and 1800MHz bands so as to make it technology neutral. Banglalink was awarded 2×5.6MHz in the 1800MHz band and 5MHz of paired spectrum in the 2100MHz band for a total fee of USD308.6 million (excluding VAT), while it will pay a further USD35 million to convert its existing spectrum

"Bangladesh raises USD1.7bn from LTE frequency tender." 15 Feb. 2018, https://www.telegeography.com/products/commsupdate/articles/..

The Rules of the Game Matter

Bangladesh

India

Bangladesh raises USD1.7bn from LTE
freq

15 Feb 201

Bangla

The Bangla
auction, far
sale bough
was 'not ha
spectrum a

Market lea
convert its
was award
of USD308.

"Bangladesh r
frequency ten
www.telegeog
supdate/articl

Airwaves in the 2,300 megahertz band sold out as telecom operators spent to increase their 4G mobile broadband services. Photo: Mint

Spectrum auction ends, govt makes Rs65,789 crore, misses target

4 min read . Updated: 07 Oct 2016, 10:08 AM IST

Upasana Jain

Proceeds from spectrum auction a fraction of the Rs5.63 trillion of airwaves on offer; no bids were received for 700 MHz, 900 Mhz bands

"Spectrum auction ends, govt makes Rs65,789 crore, misses target."
07 Oct. 2016, https://www.livemint.com/Industry/xt5r4Zs5RmzjdwuLUdwJMI/..

The Rules of the Game Matter

Bangladesh

Bangladesh raises USD1.7bn from LTE freq

15 Feb 201

Bangla

The Bangla
auction, far
sale bough
was 'not ha
spectrum a

Market lead
convert its
was award
of USD308.

"Bangladesh ra
frequency tend
www.telegeogr
supdate/article

India

"Spectrum aucti
Rs65,789 crore,
07 Oct. 2016, htt
Industry/xt5r4Zs

Ghana

MTN Ghana poised to snap up unallocated 800MHz 4G spectrum

5 Apr 2019

Ghana

Mobile network operator (MNO) MTN Ghana is lining up to purchase the two remaining 2x5MHz blocks of spectrum lots in the 800MHz band that were left unallocated after Vodafone Ghana acquired its own block of 2x5MHz for USD30 million last December, Adom News reports. 'MTN intends to acquire this remaining spectrum to enable it to continue to give its customers an increasingly better experience on the network,' MTN Corporate Services Executive Robert Kuzoe confirmed to Adom News in response to a questionnaire.

The MNO was precluded from the National Communications Authority (NCA's) auction of three separate 2x5MHz spectrum lots in the 800MHz band at the end of last year, on the grounds that it had already acquired a 2x10MHz lot in the same band back in December 2015. While the NCA confirmed at the end of the 2018 spectrum auction that 'two companies submitted applications, with Vodafone emerging as the only successful applicant,' the

"MTN Ghana poised to snap up unallocated 800MHz 4G spectrum." 05 April. 2019, https://www.telegeography.com/products/commsupdate/..

Empirical Mechanism Design

The space of all possible mechanisms is too vast!

Empirical Mechanism Design

The space of all possible mechanisms is too vast!

We focus on optimizing the parameters of an existing mechanism.

Empirical Mechanism Design

The space of all possible mechanisms is too vast!

We focus on optimizing the parameters of an existing mechanism.

How should a **mechanism designer** set **parameters** of a mechanism, given access only to **data** (or to a simulator capable of generating data) about the **game** under different choices of parameters?

Empirical Mechanism Design

The space of all possible mechanisms is too vast!

We focus on optimizing the parameters of an existing mechanism.

How should a **mechanism designer** set **parameters** of a mechanism, given access only to **data** (or to a simulator capable of generating data) about the **game** under different choices of parameters?

e.g., How should an **auctioneer** set the **reserve prices** of an auction given access only to auction log **data under different choices of reserve prices**?

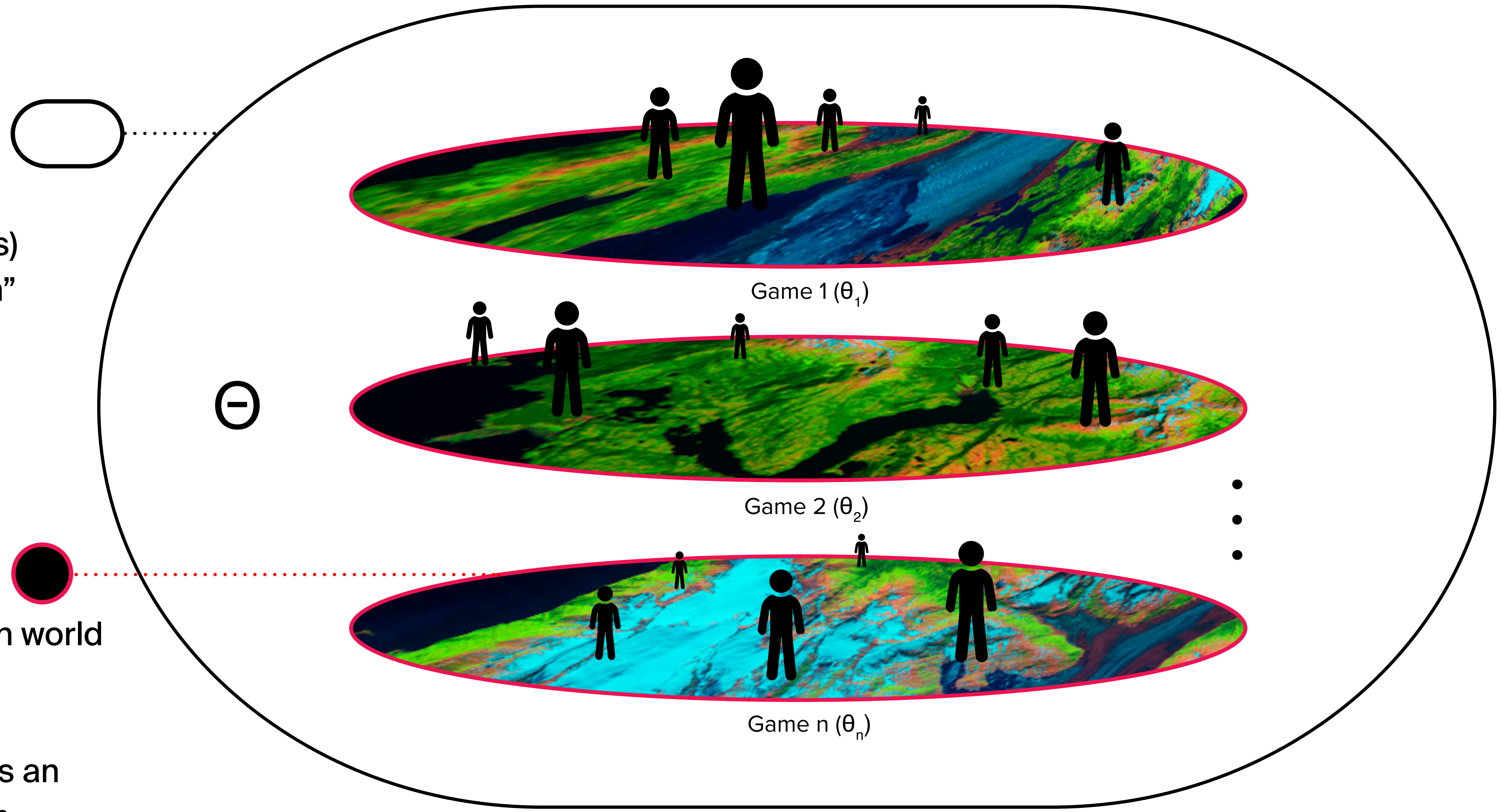
Empirical Mechanism Design - Schematic

Parameter search

— A set Θ describes all possible worlds (games)
— θ in Θ is a “mechanism”

Equilibria estimation

— Multiple players in each world
— Find an equilibria with chosen θ
— The designer measures an objective at equilibrium



Empirical Mechanism Design - Notation

Fix some parametrizable mechanism, (e.g., a first-price auction).

Empirical Mechanism Design - Notation

Fix some parametrizable mechanism, (e.g., a first-price auction).

Θ is the space of the mechanism's parameters (e.g., reserve prices)

Empirical Mechanism Design - Notation

Fix some parametrizable mechanism, (e.g., a first-price auction).

Θ is the space of the mechanism's parameters (e.g., reserve prices)

$\theta \in \Theta$ is a choice of parameters (e.g., a reserve price of $10)

Empirical Mechanism Design - Notation

Fix some parametrizable mechanism, (e.g., a first-price auction).

$\boldsymbol{\Theta}$ is the space of the mechanism's parameters (e.g., reserve prices)

$\theta \in \boldsymbol{\Theta}$ is a choice of parameters (e.g., a reserve price of \$10)

$\Gamma_\theta = \left\langle P, \mathbf{S}_\theta, u_\theta(\,\cdot\,) \right\rangle$ is a θ-simulation-based game.

Empirical Mechanism Design - Notation

Fix some parametrizable mechanism, (e.g., a first-price auction).

Θ is the space of the mechanism's parameters (e.g., reserve prices)

$\theta \in \Theta$ is a choice of parameters (e.g., a reserve price of \$10)

$\Gamma_\theta = \left\langle P, \mathbf{S}_\theta, u_\theta(\,\cdot\,) \right\rangle$ is a θ-simulation-based game.

Empirical Mechanism Design - Notation

Fix some parametrizable mechanism, (e.g., a first-price auction).

Θ is the space of the mechanism's parameters (e.g., reserve prices)

$\theta \in \Theta$ is a choice of parameters (e.g., a reserve price of \$10)

$\Gamma_\theta = \left\langle P, \mathbf{S}_\theta, u_\theta(\,\cdot\,) \right\rangle$ is a θ-simulation-based game.

Empirical Mechanism Design - Problem Statement

$f(\vec{s}; \Gamma_\theta)$ is the designer's objective function (e.g., revenue) evaluated at profile $\vec{s}$ in game Γ_θ, where $f(\vec{s}; \Gamma_\theta) \in \mathbb{R}$

Empirical Mechanism Design - Problem Statement

$f(\vec{s};\Gamma_\theta)$ is the designer's objective function (e.g., revenue) evaluated at profile $\vec{s}$ in game Γ_θ, where $f(\vec{s};\Gamma_\theta) \in \mathbb{R}$

$E(\Gamma_\theta)$ is the set of equilibria of θ-simulation-based game Γ_θ.

Empirical Mechanism Design - Problem Statement

$f(\vec{s};\Gamma_\theta)$ is the designer's objective function (e.g., revenue) evaluated at profile $\vec{s}$ in game Γ_θ, where $f(\vec{s};\Gamma_\theta) \in \mathbb{R}$

$E(\Gamma_\theta)$ is the set of equilibria of θ-simulation-based game Γ_θ.

Since (often) it is the case that equilibria are not unique ($|E(\Gamma_\theta)| > 1$), define:

Empirical Mechanism Design - Problem Statement

$f(\vec{s};\Gamma_\theta)$ is the designer's objective function (e.g., revenue) evaluated at profile $\vec{s}$ in game Γ_θ, where $f(\vec{s};\Gamma_\theta) \in \mathbb{R}$

$E(\Gamma_\theta)$ is the set of equilibria of θ-simulation-based game Γ_θ.

Since (often) it is the case that equilibria are not unique ($|E(\Gamma_\theta)| > 1$), define:

$$F(\theta;\Gamma_\theta) = \min_{\vec{s}\in E(\Gamma_\theta)} f(\vec{s};\Gamma_\theta)$$

Empirical Mechanism Design - Problem Statement

$f(\vec{s}; \Gamma_\theta)$ is the designer's objective function (e.g., revenue) evaluated at profile $\vec{s}$ in game Γ_θ, where $f(\vec{s}; \Gamma_\theta) \in \mathbb{R}$

$E(\Gamma_\theta)$ is the set of equilibria of θ-simulation-based game Γ_θ.

Since (often) it is the case that equilibria are not unique ($|E(\Gamma_\theta)| > 1$), define:

Worst-case, could also define average- or best-case.

$$F(\theta; \Gamma_\theta) = \min_{\vec{s} \in E(\Gamma_\theta)} f(\vec{s}; \Gamma_\theta)$$

Empirical Mechanism Design - Problem Statement

$f(\vec{s};\Gamma_\theta)$ is the designer's objective function (e.g., revenue) evaluated at profile $\vec{s}$ in game Γ_θ, where $f(\vec{s};\Gamma_\theta) \in \mathbb{R}$

$E(\Gamma_\theta)$ is the set of equilibria of θ-simulation-based game Γ_θ.

Since (often) it is the case that equilibria are not unique ($|E(\Gamma_\theta)| > 1$), define:

Worst-case, could also define average- or best-case.

$$F(\theta;\Gamma_\theta) = \min_{\vec{s}\in E(\Gamma_\theta)} f(\vec{s};\Gamma_\theta)$$

The mechanism designer's problem is to find θ^* such that:

$$\theta^* \in \arg\max_{\theta\in\Theta} F(\theta;\Gamma_\theta)$$

Empirical Mechanism Design - Problem Statement

$f(\vec{s}; \Gamma_\theta)$ is the designer's objective function (e.g., revenue) evaluated at profile $\vec{s}$ in game Γ_θ, where $f(\vec{s}; \Gamma_\theta) \in \mathbb{R}$

$E(\Gamma_\theta)$ is the set of equilibria of θ-simulation-based game Γ_θ.

Since (often) it is the case that equilibria are not unique ($|E(\Gamma_\theta)| > 1$), define:

Worst-case, could also define average- or best-case.

$$F(\theta; \Gamma_\theta) = \min_{\vec{s} \in E(\Gamma_\theta)} f(\vec{s}; \Gamma_\theta)$$

The mechanism designer's problem is to find θ^* such that:

$$\theta^* \in \arg\max_{\theta \in \Theta} F(\theta; \Gamma_\theta)$$

$|\Theta| < \infty$, PAC algorithm
$|\Theta| = \infty$, Bayesian Optimization

The "Design" Plan (a.k.a. Outline Part 3)

- Empirical Mechanism Design
- Experiments - Ad Auctions

The "Design" Plan (a.k.a. Outline Part 3)

- Empirical Mechanism Design
- Experiments - Ad Auctions

The "Design" Plan (a.k.a. Outline Part 3)

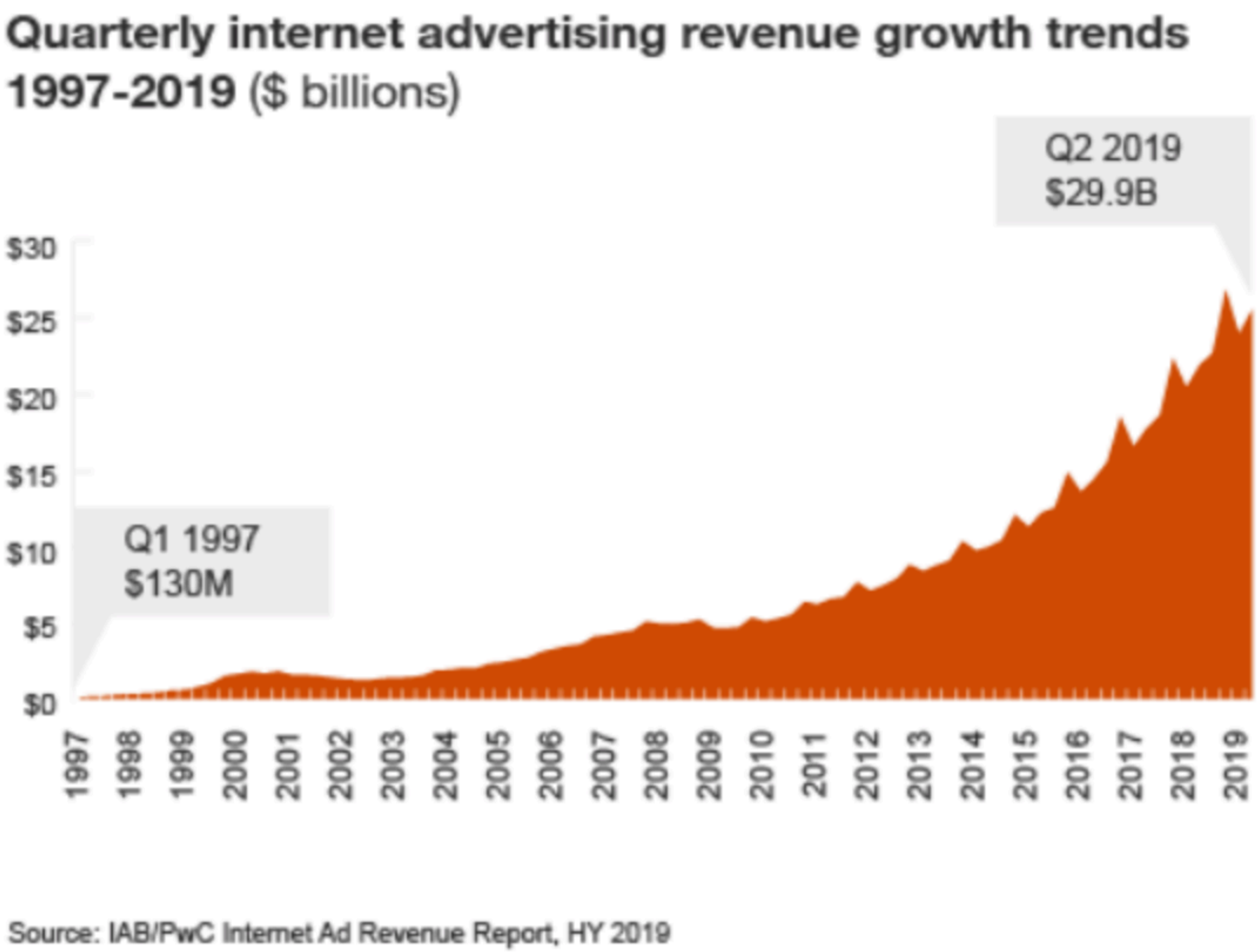
- ~~Empirical Mechanism Design~~
- Experiments - Ad Auctions

Electronic Advertisement Auctions

Electronic Advertisement Auctions

Image Credit: webcapitaltop.com/servicios/campanas-display

Electronic Advertisement Auctions



Electronic Advertisement Exchanges

- At the heart of electronic advertisement are **ad-exchanges**: centralized locations that match supply to demand, typically though some kind of auction.

Electronic Advertisement Exchanges

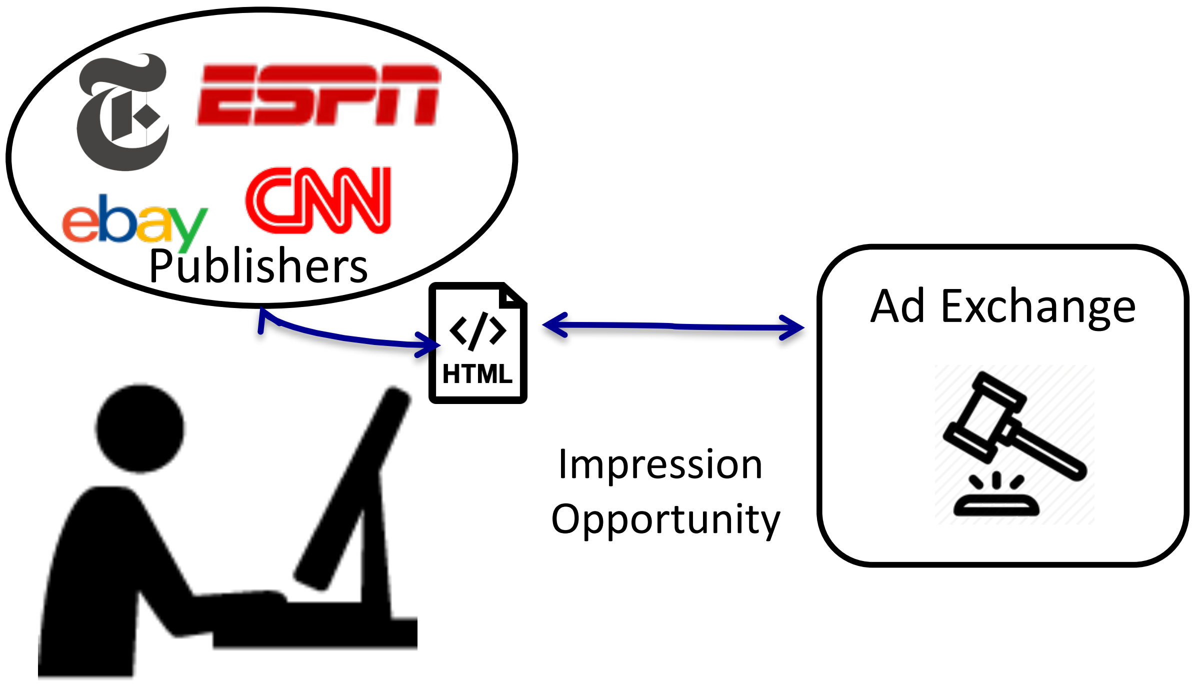
- At the heart of electronic advertisement are **ad-exchanges**: centralized locations that match supply to demand, typically though some kind of auction.
- Advertisers might have different **objectives**, e.g., to immediately convert clicks into purchases, or to maintain brand awareness.

Electronic Advertisement Exchanges

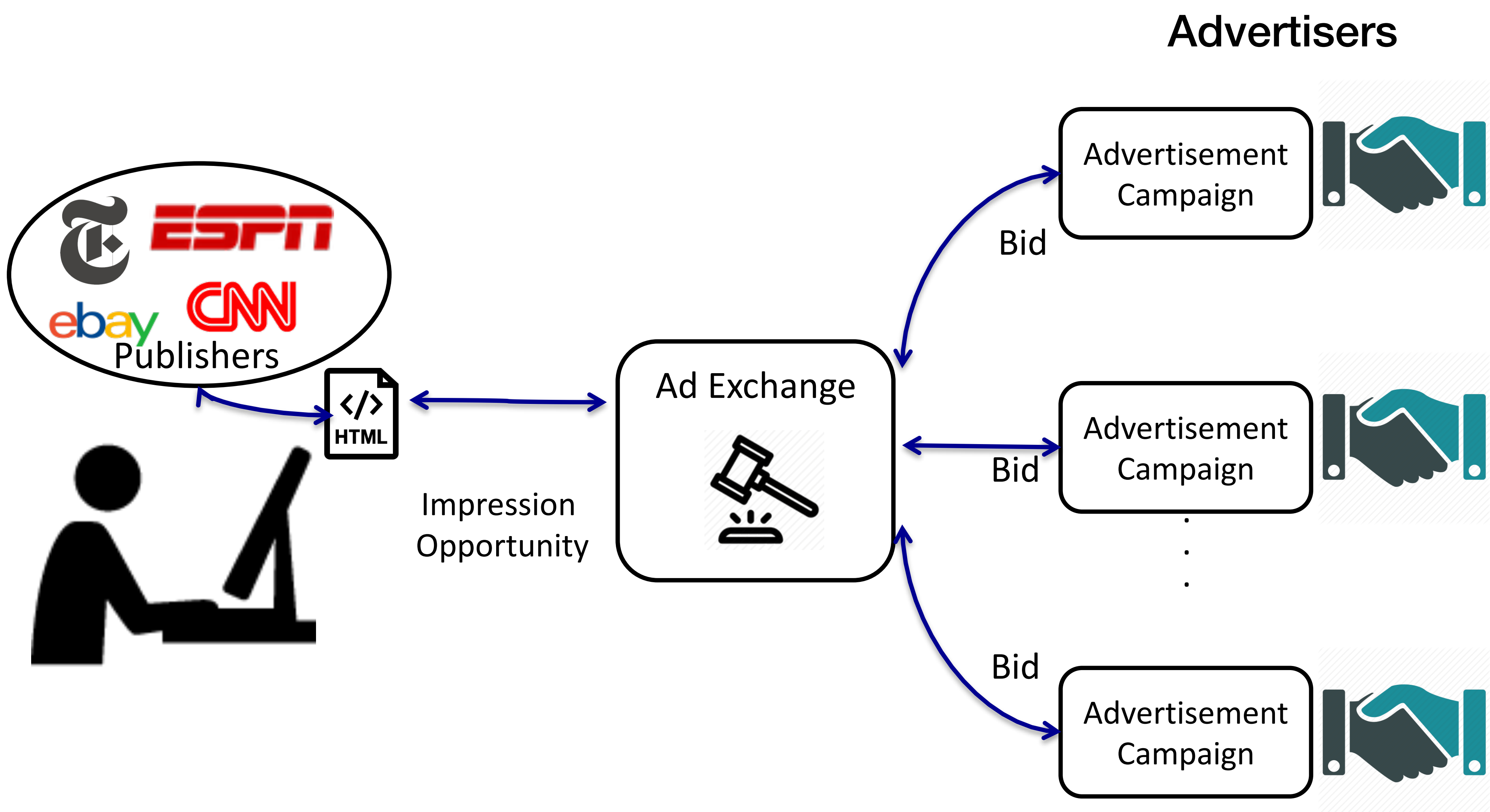
- At the heart of electronic advertisement are **ad-exchanges**: centralized locations that match supply to demand, typically though some kind of auction.
- Advertisers might have different **objectives**, e.g., to immediately convert clicks into purchases, or to maintain brand awareness.
- We focus on **brand-awareness advertisement** where advertisers need to reach a certain number of potential customers, from certain demographics, for a fixed (pre-determined) budget

Electronic Advertisement Exchanges - Schematic

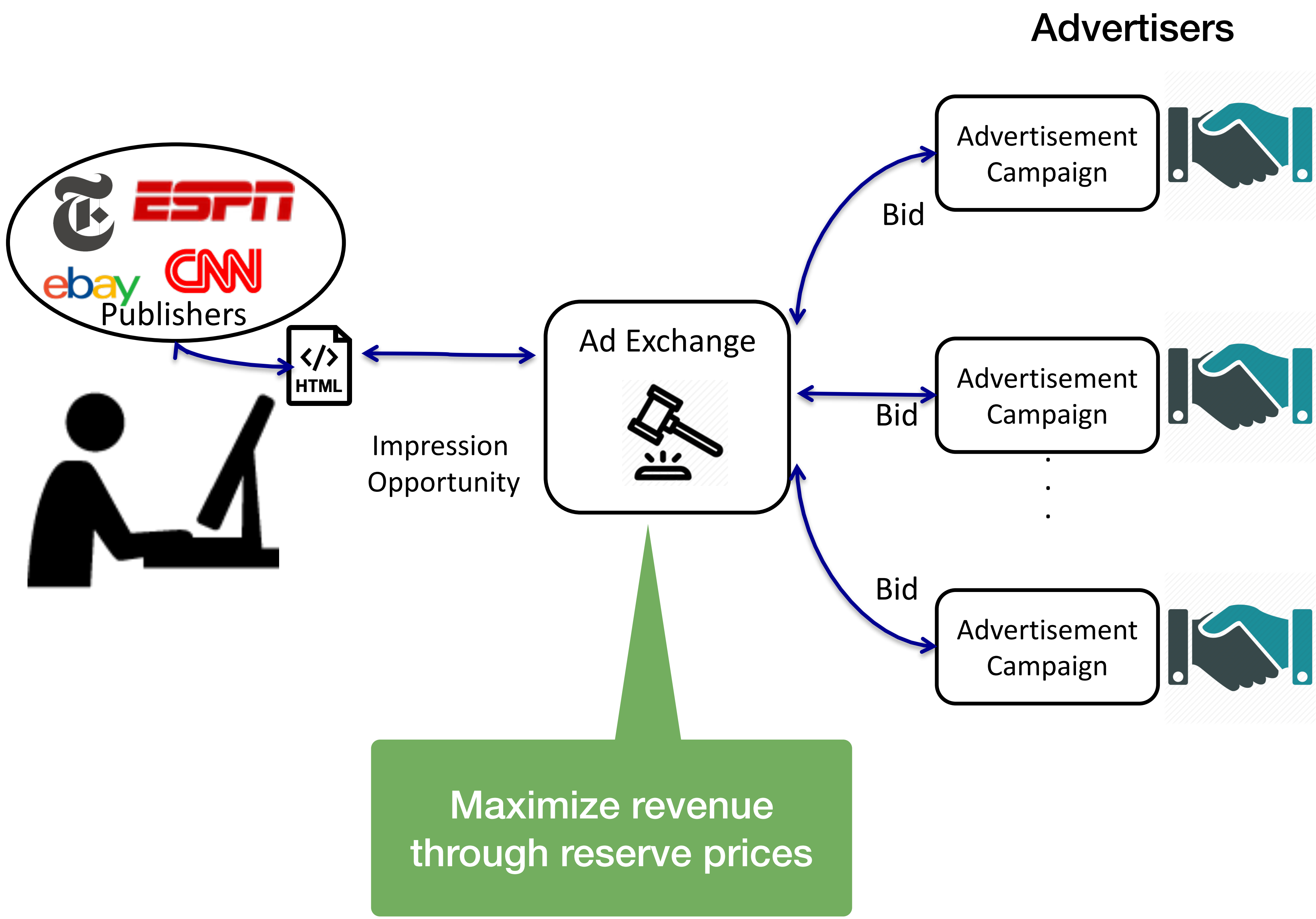
Electronic Advertisement Exchanges - Schematic



Electronic Advertisement Exchanges - Schematic



Electronic Advertisement Exchanges - Schematic



Electronic Advertisement Exchanges - Model

- **Stage 1**: the *ad exchange* announces $\vec{r} \in \mathbb{R}_{+}^{m}$, where $< r_1, \ldots, r_m > \in \Theta$ is such that r_j is the reserve price for the j^{th} demographic or market segment.

Electronic Advertisement Exchanges - Model

- **Stage 1**: the *ad exchange* announces $\vec{r} \in \mathbb{R}^m_+$, where $< r_1, \ldots, r_m > \in \Theta$ is such that r_j is the reserve price for the j^{th} demographic or market segment.

- **Stage 2**: *all agents* submit their bids (produced by heuristic strategies we outline in the next slide).

Electronic Advertisement Exchanges - Model

- **Stage 1**: the *ad exchange* announces $\vec{r} \in \mathbb{R}_+^m$, where $< r_1, \ldots, r_m > \in \Theta$ is such that r_j is the reserve price for the j^{th} demographic or market segment.

- **Stage 2**: *all agents* submit their bids (produced by heuristic strategies we outline in the next slide).

- **Stage 3**: some fixed number of *impression opportunities* arrive, where the demographic of each is drawn from some distribution.

Electronic Advertisement Exchanges - Model

- **Stage 1**: the *ad exchange* announces $\vec{r} \in \mathbb{R}^m_+$, where $< r_1, \ldots, r_m > \in \Theta$ is such that r_j is the reserve price for the j^{th} demographic or market segment.

- **Stage 2**: *all agents* submit their bids (produced by heuristic strategies we outline in the next slide).

- **Stage 3**: some fixed number of *impression opportunities* arrive, where the demographic of each is drawn from some distribution.

- **Stage 4**: for each impression opportunity, the auctioneer runs an auction. Final *allocation and payments* are computed.

Electronic Advertisement Exchanges - Model

- **Stage 1**: the *ad exchange* announces $\vec{r} \in \mathbb{R}_+^m$, where $< r_1, \ldots, r_m > \in \Theta$ is such that r_j is the reserve price for the j^{th} demographic or market segment.

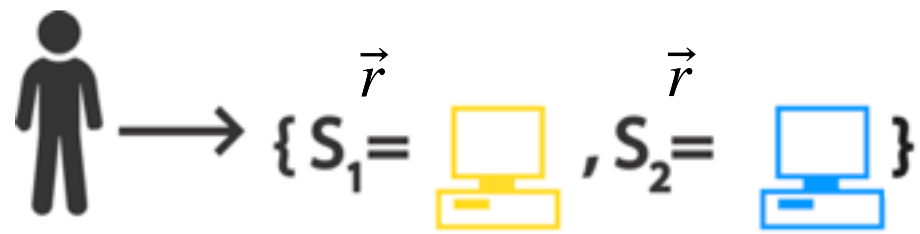
- **Stage 2**: *all agents* submit their bids (produced by heuristic strategies we outline in the next slide).

- **Stage 3**: some fixed number of *impression opportunities* arrive, where the demographic of each is drawn from some distribution.

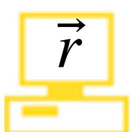
- **Stage 4**: for each impression opportunity, the auctioneer runs an auction. Final *allocation and payments* are computed.

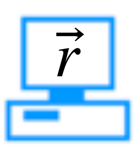
- **Input**: $\vec{r}$, **Output**: ad exchange revenue (sum of all payments).

Electronic Advertisement Exchanges - Heuristics

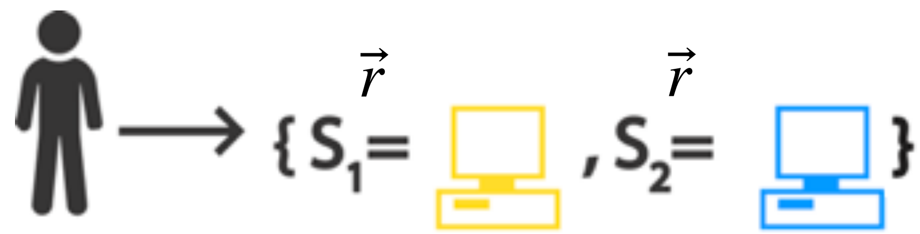


We devised two heuristics for our experimental setup.

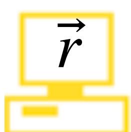




Electronic Advertisement Exchanges - Heuristics

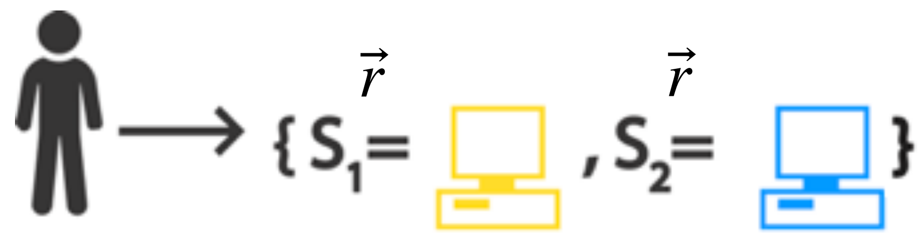


We devised two heuristics for our experimental setup.



Walrasian or (Competitive) Equilibrium, denote by ***WE***

Electronic Advertisement Exchanges - Heuristics

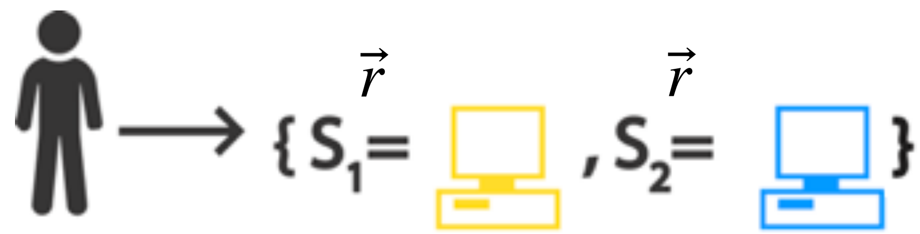


We devised two heuristics for our experimental setup.

 Walrasian or (Competitive) Equilibrium, denote by ***WE***

- Bidding based on an (approximate) competitive equilibrium.

Electronic Advertisement Exchanges - Heuristics

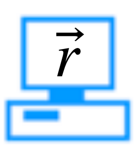


We devised two heuristics for our experimental setup.

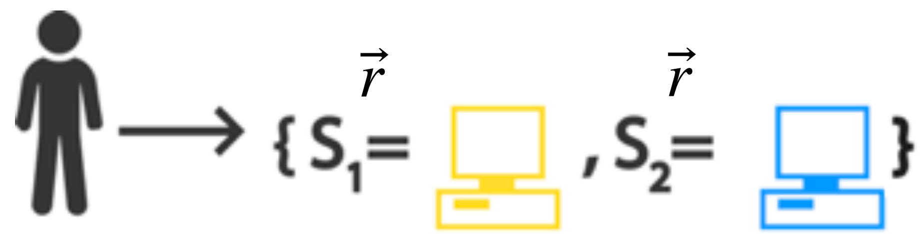
Walrasian or (Competitive) Equilibrium, denote by ***WE***

- Bidding based on an (approxima… m.

Approximating equilibria in combinatorial markets, work of interest in its own right, (Areyan et al. AAMAS17)



Electronic Advertisement Exchanges - Heuristics



We devised two heuristics for our experimental setup.

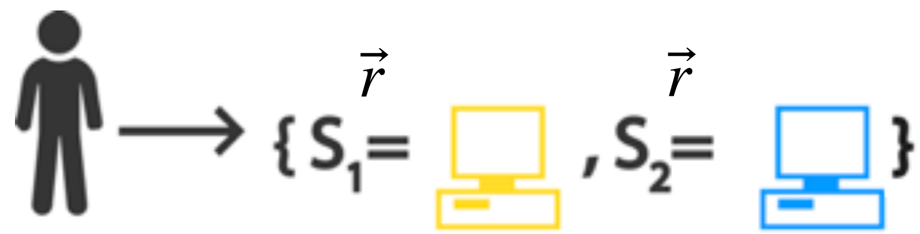
$\vec{r}$ Walrasian or (Competitive) Equilibrium, denote by ***WE***

- Bidding based on an (approxima … n.

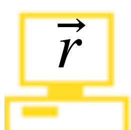
Approximating equilibria in combinatorial markets, work of interest in its own right, (Areyan et al. AAMAS17)

$\vec{r}$ Waterfall, denoted by ***WF***

Electronic Advertisement Exchanges - Heuristics

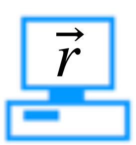


We devised two heuristics for our experimental setup.


Walrasian or (Competitive) Equilibrium, denote by ***WE***

- Bidding based on an (approxima … m.

Approximating equilibria in combinatorial markets, work of interest in its own right, (Areyan et al. AAMAS17)


Waterfall, denoted by ***WF***

- Bidding based on simulating the ad exchange dynamics.

Experimental Setup

- Draw $K = 500$ *impression opportunities* distributed in 8 market segments.

Experimental Setup

- Draw $K = 500$ *impression opportunities* distributed in 8 market segments.
- We experiment with $N = 4$ *agents*, each allowed to chose from the two strategies mentioned before, i.e., S = {***WE***, ***WF***}.

Experimental Setup

- Draw $K = 500$ *impression opportunities* distributed in 8 market segments.

- We experiment with $N = 4$ *agents*, each allowed to chose from the two strategies mentioned before, i.e., S = {***WE***, ***WF***}.

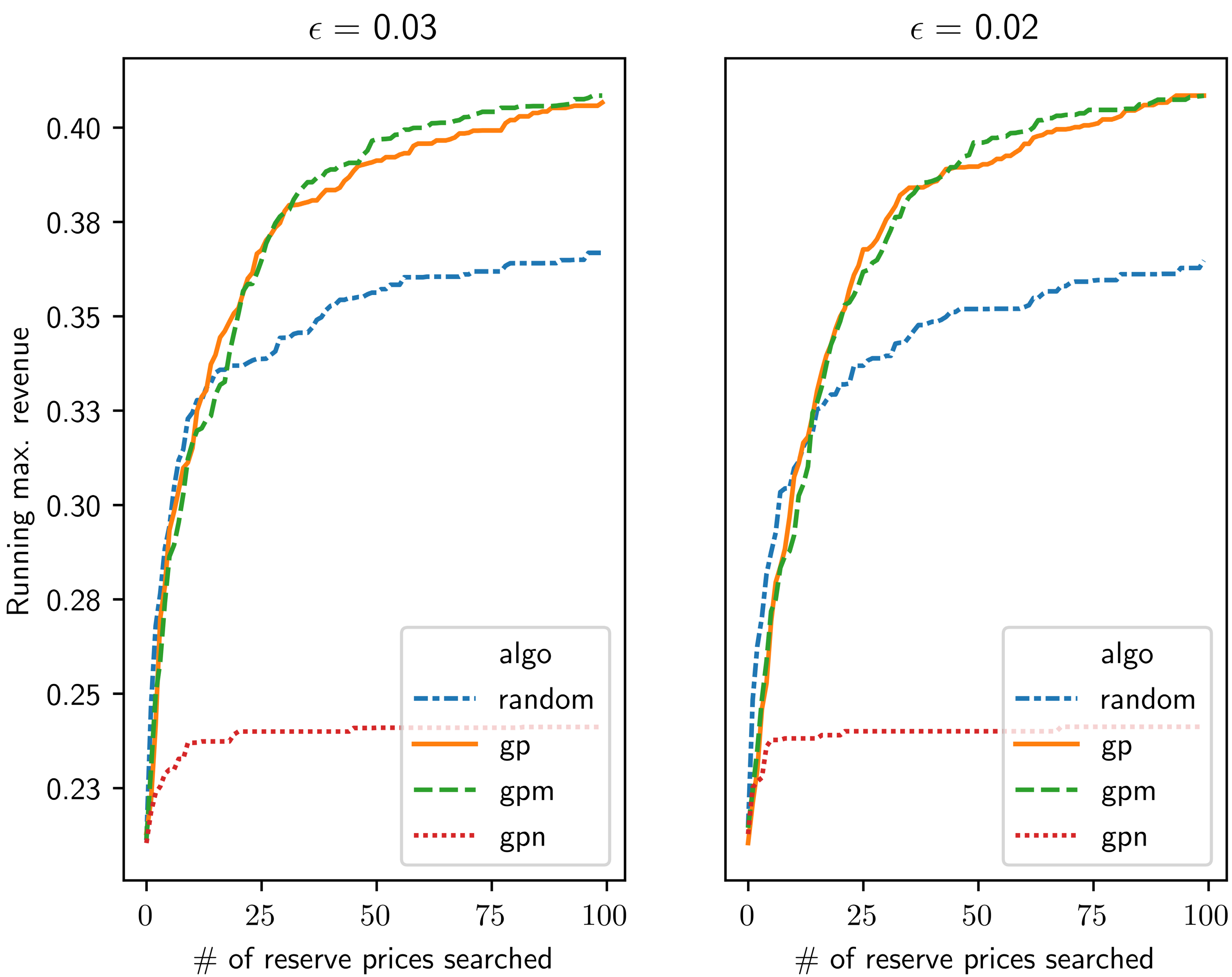
- Design space is $\Theta = \mathbb{R}^8$. Here, $< r_1, \ldots, r_8 > \in \Theta$ is such that r_j is the reserve price for the j^{th} demographic or market segment.

Experimental Setup

- Draw $K = 500$ *impression opportunities* distributed in 8 market segments.
- We experiment with $N = 4$ *agents*, each allowed to chose from the two strategies mentioned before, i.e., S = {***WE***, ***WF***}.
- Design space is $\Theta = \mathbb{R}^8$. Here, $< r_1, \ldots, r_8 > \in \Theta$ is such that r_j is the reserve price for the j^{th} demographic or market segment.
- The task then if to find an 8-dimensional vector of reserve prices $\vec{r}^* \in \Theta$ that maximizes the ad exchange revenue, at equilibrium.

Experimental Results

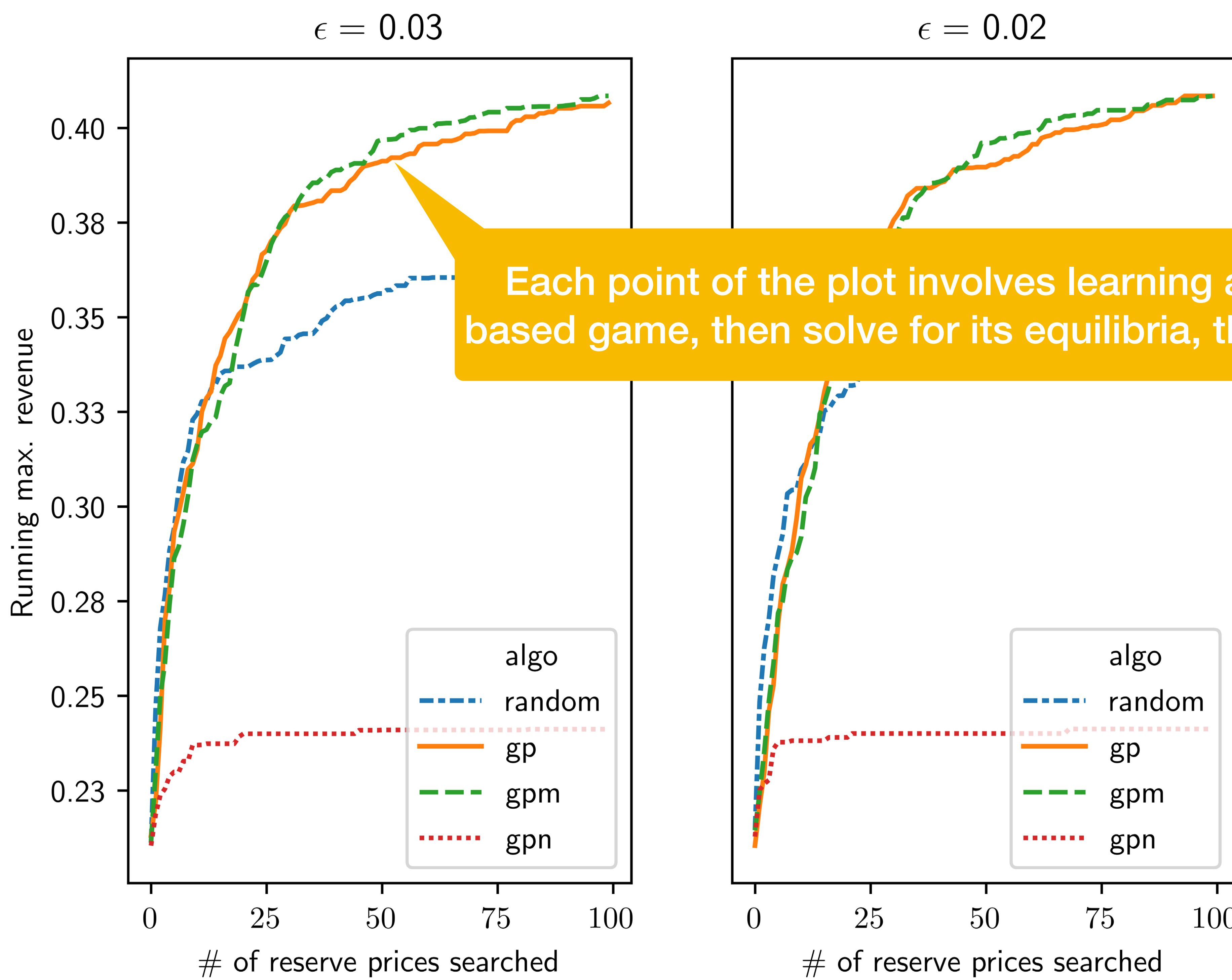
$\delta = 0.1$



All code available at github.com/eareyan/emd-adx

Experimental Results

$\delta = 0.1$



Each point of the plot involves learning a simulation-based game, then solve for its equilibria, then compute F

All code available at github.com/eareyan/emd-adx

The "Design" Plan (a.k.a. Outline Part 3)

- ~~Empirical Mechanism Design~~
- Experiments - Ad Auctions

The "Design" Plan (a.k.a. Outline Part 3)

- ~~Empirical Mechanism Design~~
- ~~Experiments - Ad Auctions~~

Summary Part 3

- We contribute an end-to-end methodology for the optimization of mechanisms' parameters.

Summary Part 3

- We contribute an end-to-end methodology for the optimization of mechanisms' parameters.
- We empirically showed the effectiveness of our BO algorithms in a styled but rich simulation of electronic advertisement exchanges.

Acknowledgments

Amy Greenwald

Collaborators

Cyrus Cousins

Yasser Mohammad

Marilyn George

Denizalp Goktas

Institutions and Mentors

UNIVERSIDAD CENTRAL CARACAS - VENEZUELA

Ignacio Calderón

INDIANA UNIVERSITY BLOOMINGTON

Esfandiar Haghverdi

amazon

Jayash Koshal

venezuelatuya.com

Hernan Rosas

Family

Thank you!