Tractable Computational Methods for Finding Nash Equilibria of Perfect-Information Position Auctions

David Robert Martin Thompson
Kevin Leyton-Brown

Department of Computer Science
University of British Columbia
{daveth|kevinlb}@cs.ubc.ca

Motivation

How will bidders behave in a position auction
that does not meet the assumptions
for which theoretical results are known?

Our approach: compute Nash equilibrium

Main hurdle: existing algorithms work with
normal form; infeasibly large for ad auctions

Main message: preliminary, but it works

Outline

- Auctions & Model
- Action-Graph Games
- Auctions as AGGs
- Computational Experiments
- Economic Experiments

Types of Position Auctions

- Dimensions:
  - Generalized First Price vs. Generalized Second Price
  - Pay-per-click vs. Pay-per-impression
  - Weighted vs. Unweighted:
    - “Effective Bid”: bid * weight
    - Ads ranked by effective bid
    - Payment: effective bid / weight
- Current Usage (Google,Microsoft,Yahoo!):
  - Weighted, Per-Click, GSP

Model of Auction Setting

Full-information, one-shot game [Varian, 2007; Edelman, Ostrovsky, Schwarz, 2006 ("EOS")]

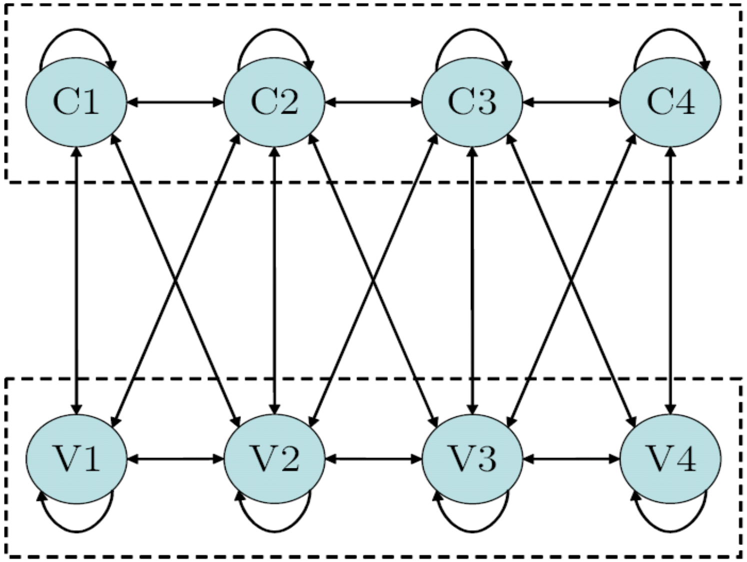
	Weights	**CTR across positions**	**CTR across bidders**	**Value per Click**	**Bid Amounts**
[EOS]	Always 1	Decreasing	Constant	One value per bidder	Continuous
[Varian]	Arbitrary	Decreasing	Proportional to Weight ("Separable")	One value per bidder	Continuous
Our model	Arbitrary	Arbitrary	Arbitrary	Arbitrary	Discrete

Outline

- Auctions & Model
- Action-Graph Games
- Auctions as AGGs
- Computational Experiments
- Economic Experiments

What are AGGs?

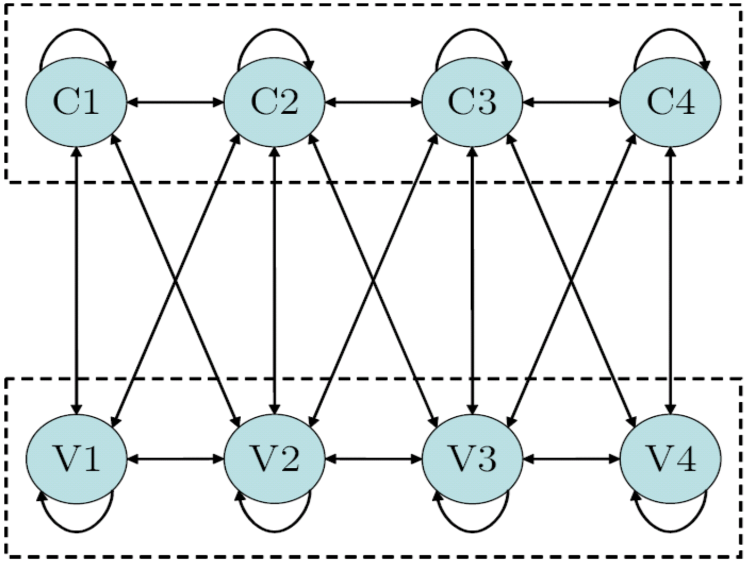
- Action Graphs:
  - Each node represents an action.
  - Arcs indicate payoff dependencies.



- [Bhat & Leyton-Brown, 2004; Jiang & Leyton-Brown, 2006]

What are AGGs?

- Action Graphs:
  - Each node represents an action.
  - Arcs indicate payoff dependencies.
  - “Function Nodes” increase sparsity.



- [Bhat & Leyton-Brown, 2004; Jiang & Leyton-Brown, 2006]

Why Use AGGs? [Bhat & Leyton-Brown, 2004]

- Small: Compact representation of a one-shot, full-information game
  - Frequently polynomial in n
- Fast: Dynamic programing can compute expected utility in ~$O(an^{i+1})$ [Jiang & Leyton-Brown, 2006]
  - Plug into existing equilibrium solvers (e.g. simplicial subdivision [van der Laan, Talman, and van Der Heyden, 1987] or GNM [Govindan, Wilson, 2003]) for exponential speedup

Outline

- Auctions & Model
- Action-Graph Games
- Auctions as AGGs
- Computational Experiments
- Economic Experiments

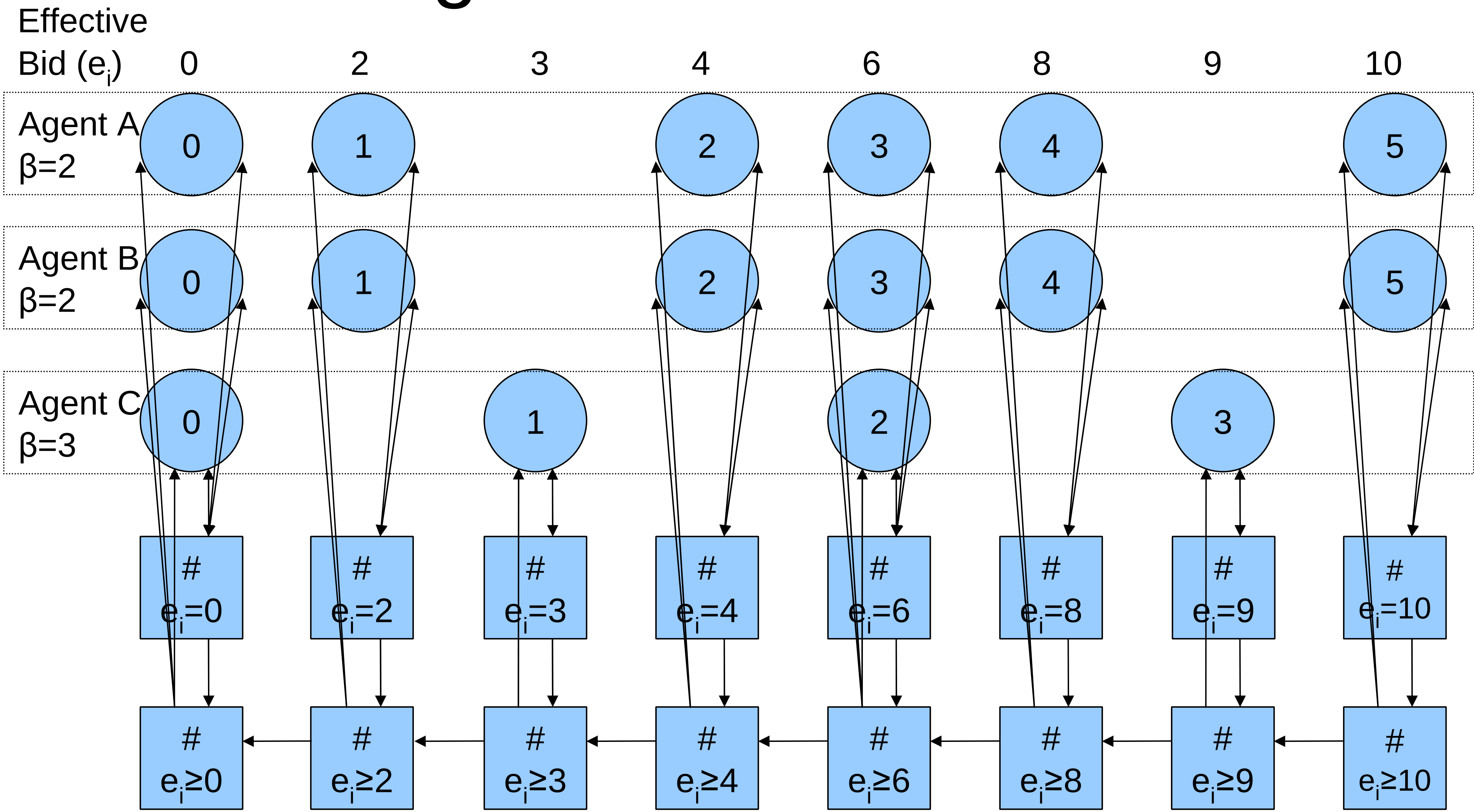
Weighted GFP as AGG

Agent A
β=2

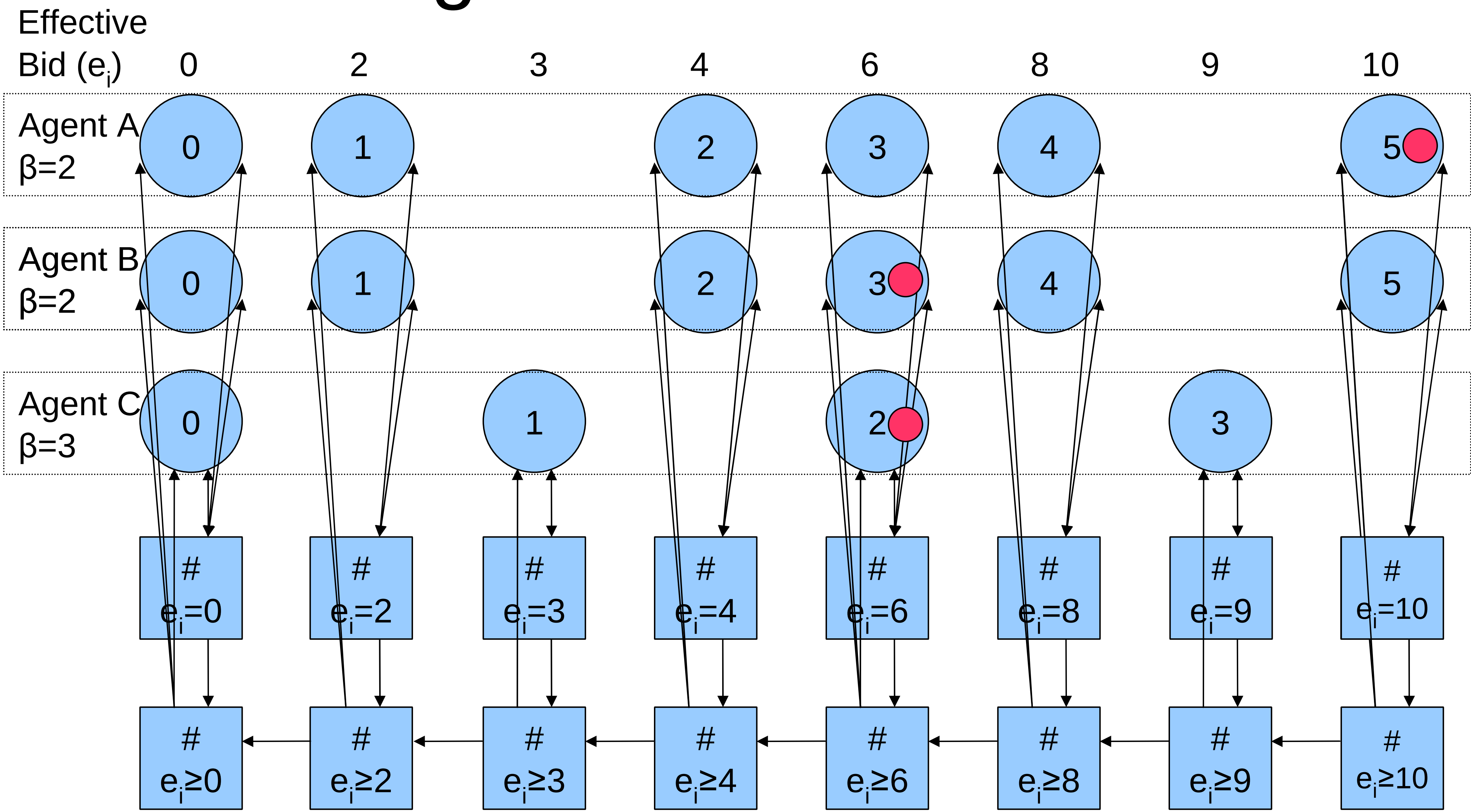
Agent B
β=2

Agent C
β=3

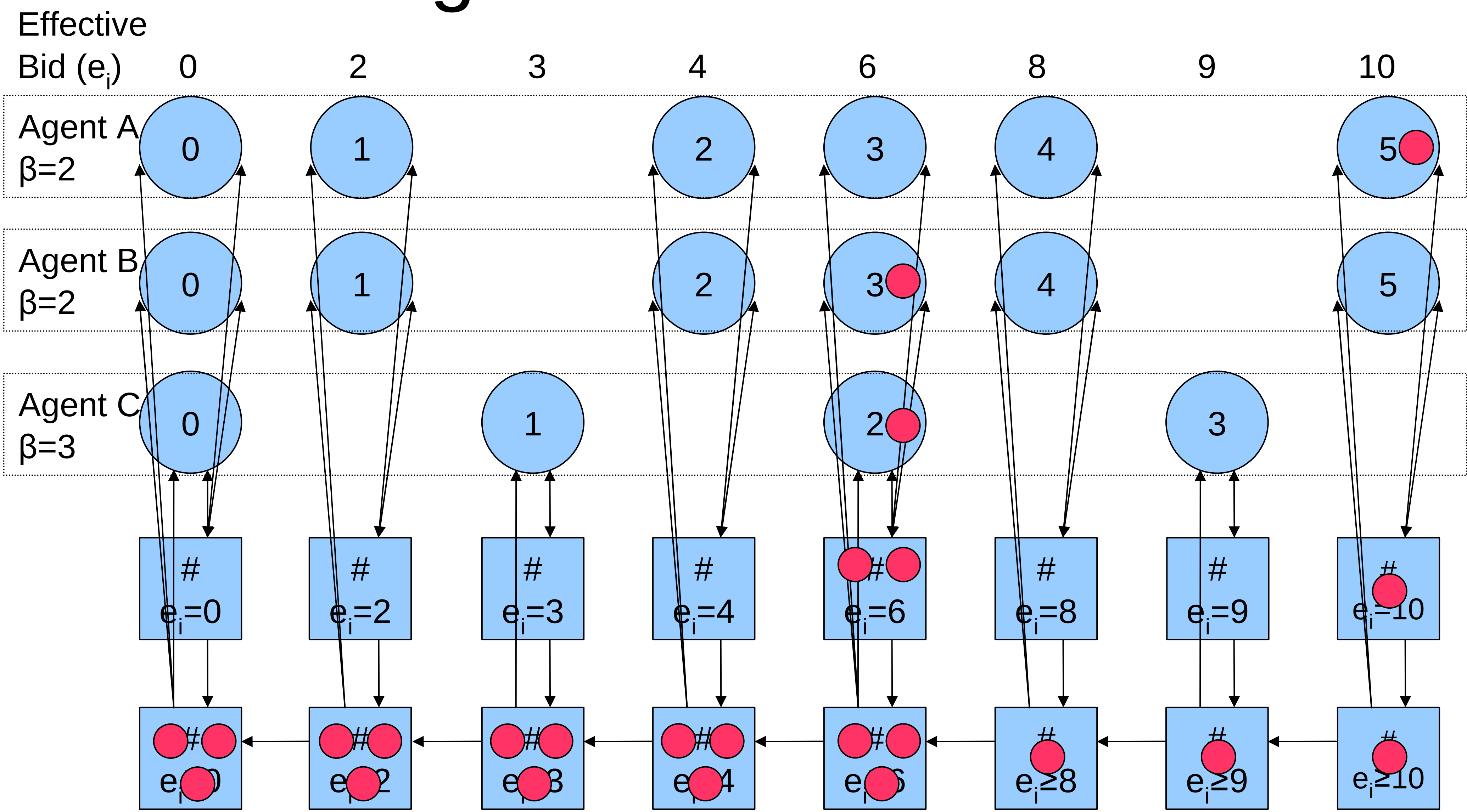
Weighted GFP as AGG



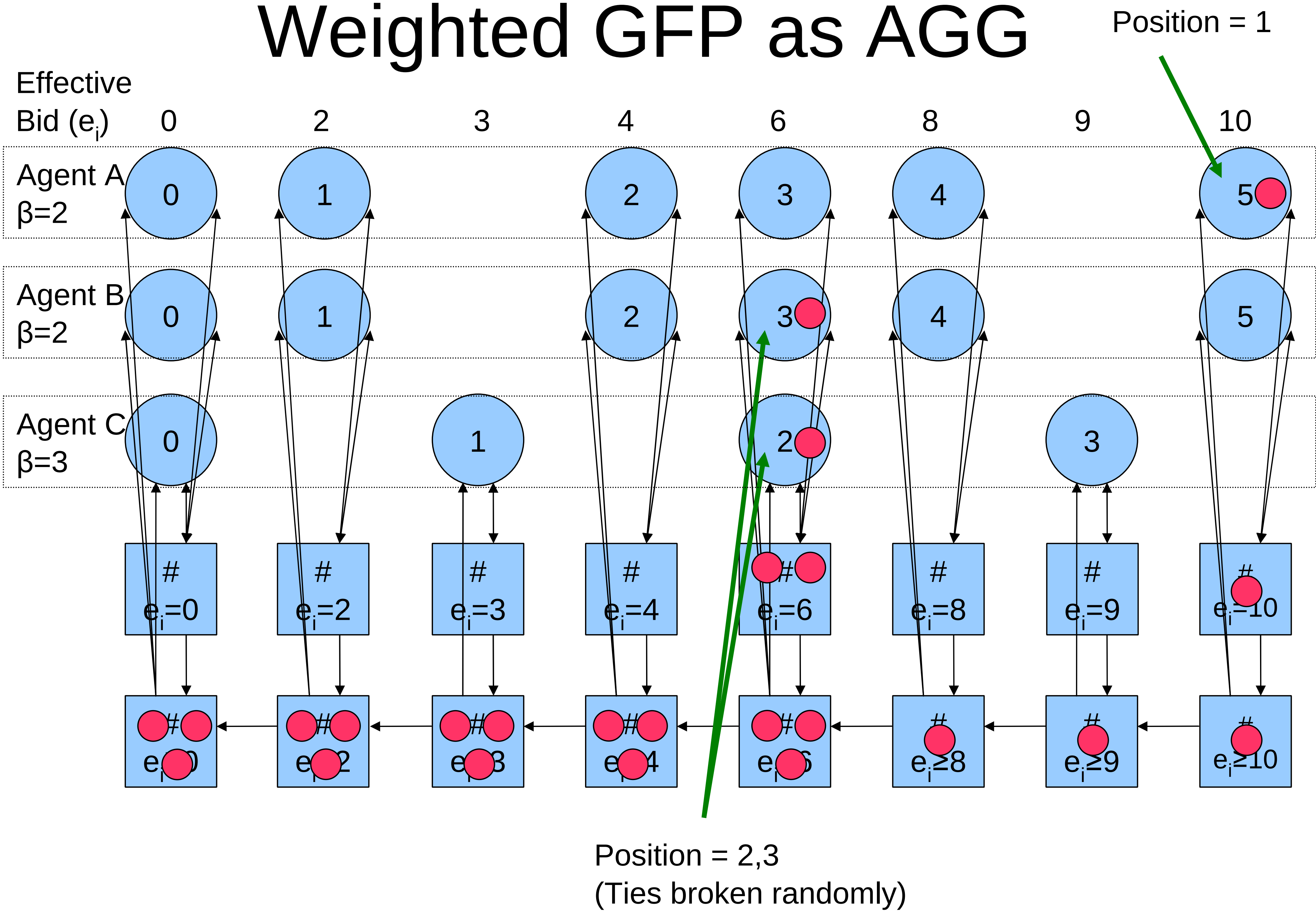
Weighted GFP as AGG



Weighted GFP as AGG



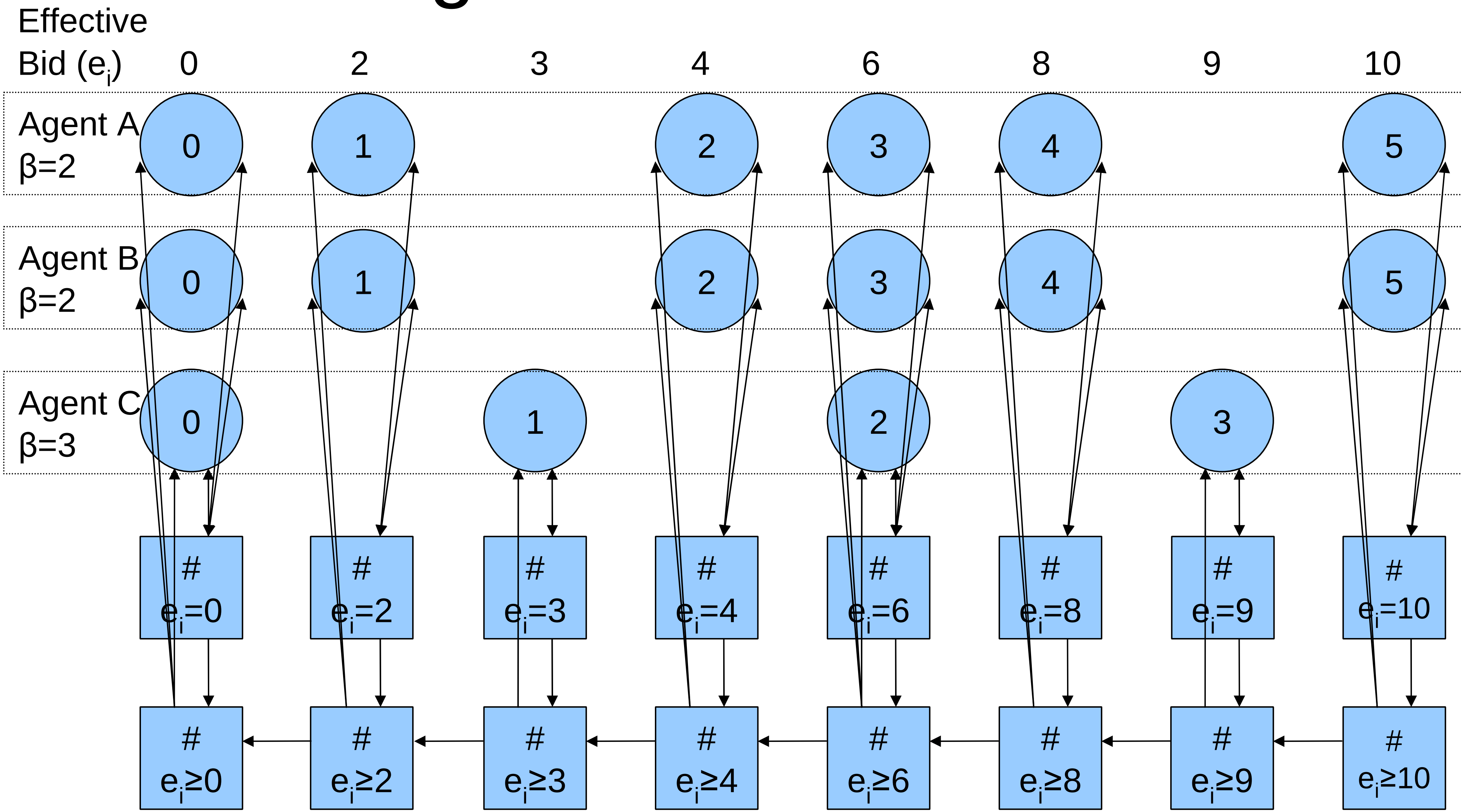
Weighted GFP as AGG



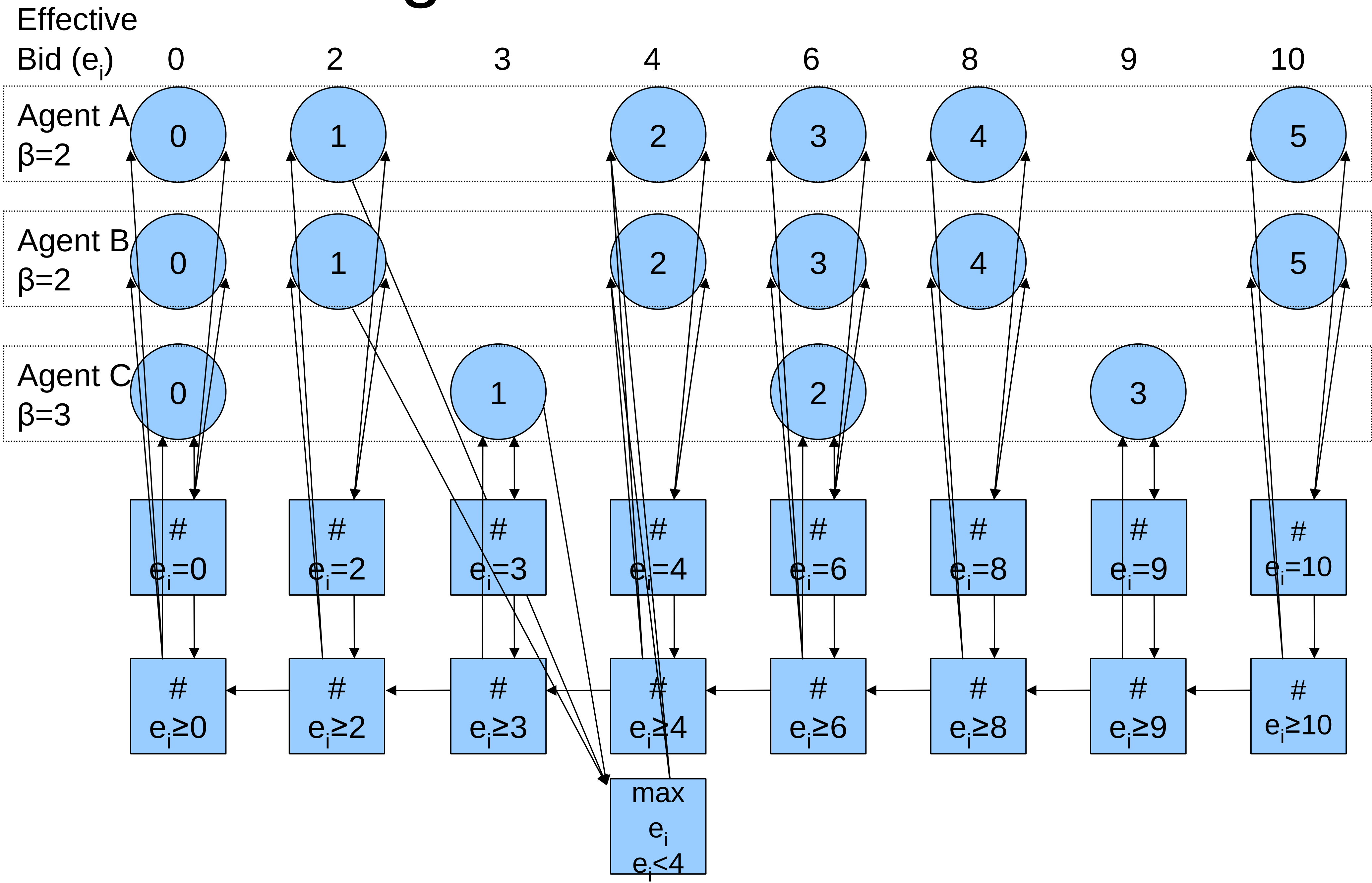
Representing GSP

- Start from a GFP graph
  - same method of computing a bidder’s position
- We need to add new nodes to compute prices

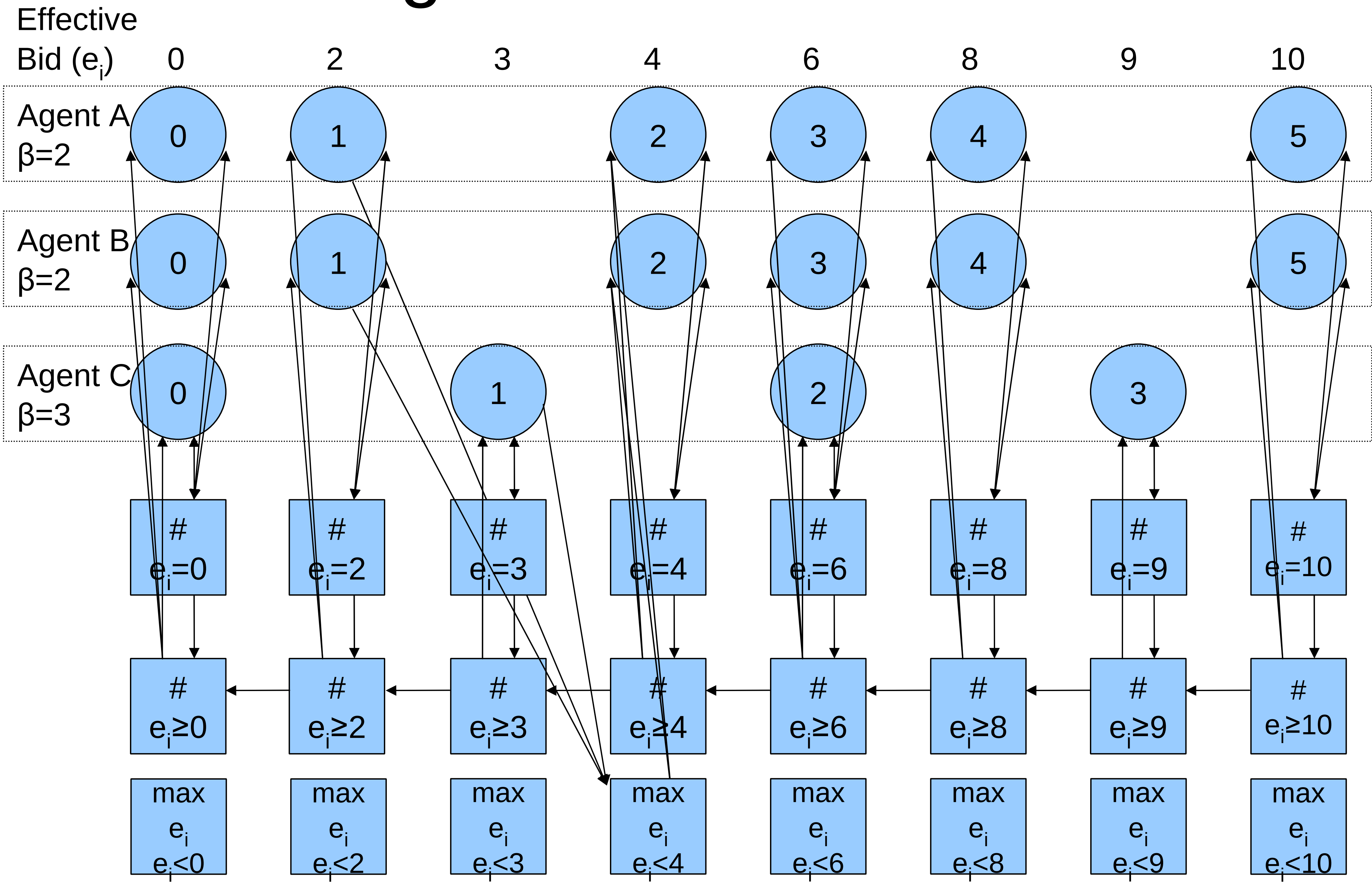
Weighted GSP as AGG



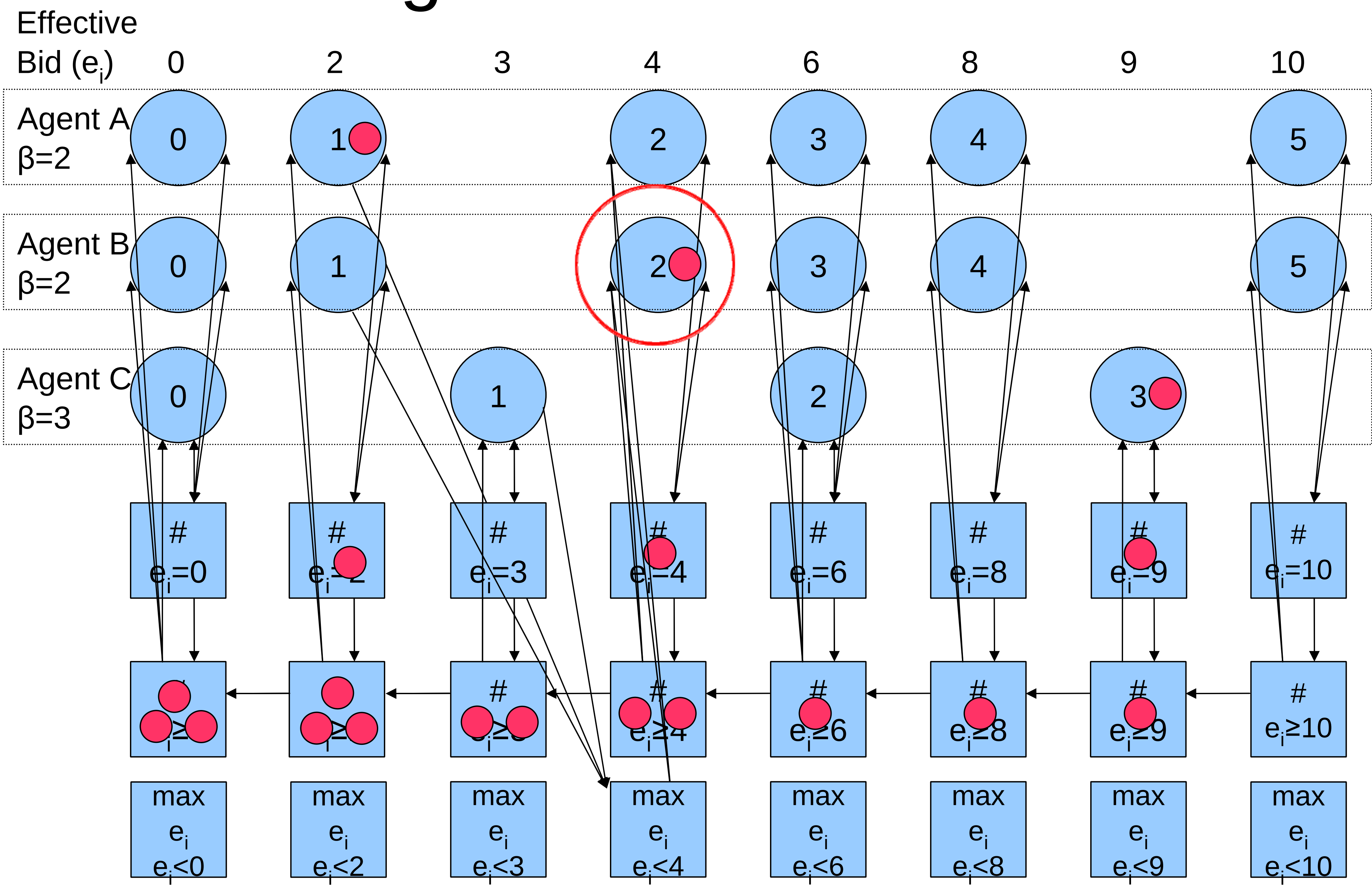
Weighted GSP as AGG



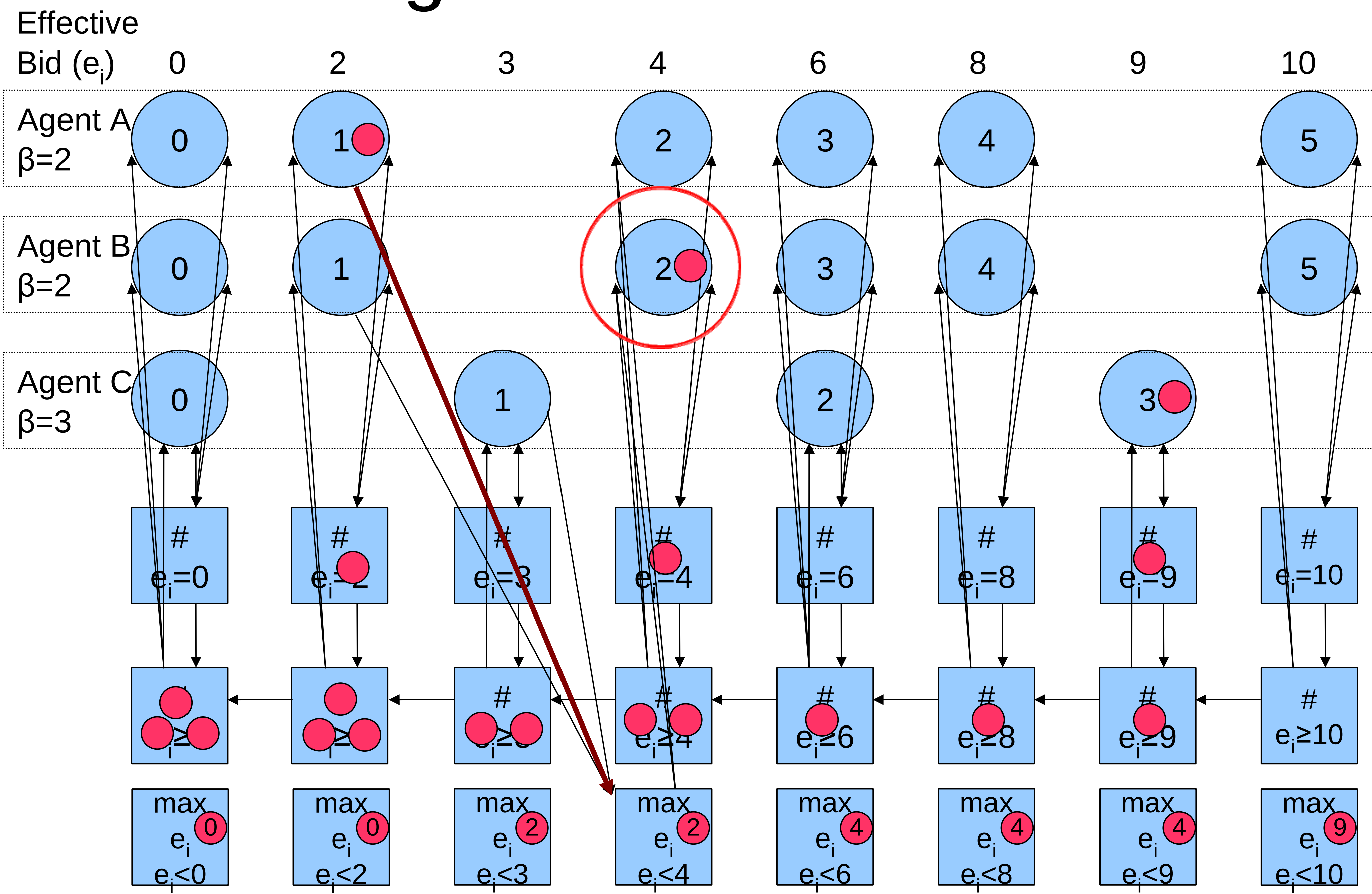
Weighted GSP as AGG



Weighted GSP as AGG

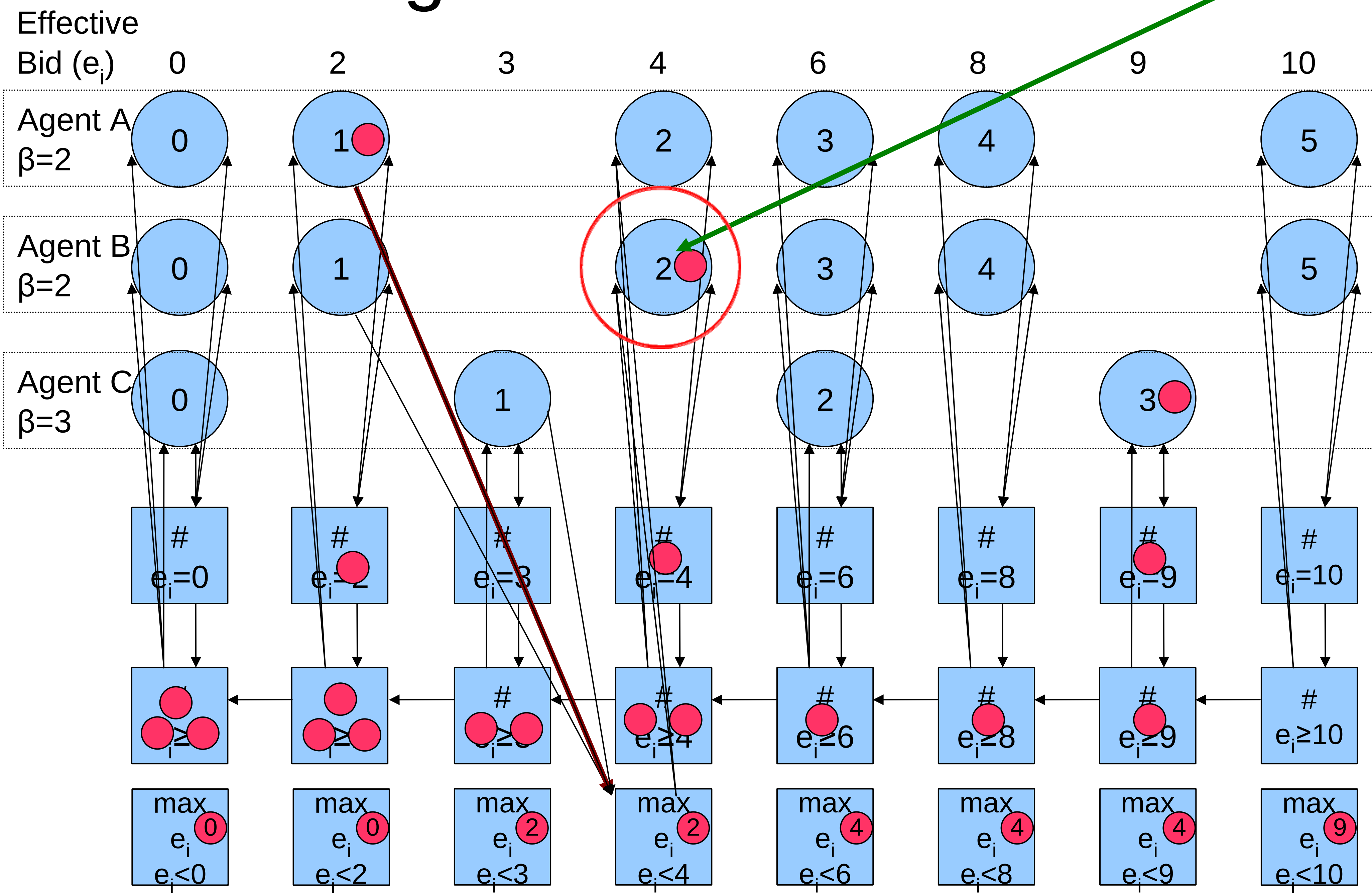


Weighted GSP as AGG



Weighted GSP as AGG

Position = 2
Price = 2/β=1



Outline

- Auctions & Model
- Action-Graph Games
- Auctions as AGGs
- Computational Experiments
- Economic Experiments

Model of Auction Setting

	Weights	CTR across positions	CTR across bidders	Value per Click	Bid Amounts
[EOS]	Always 1	Decreasing	Constant	One value per bidder	Continuous
[Varian]	Arbitrary	Decreasing	Proportional to Weight ("Separable")	One value per bidder	Continuous
Our model	Arbitrary	Arbitrary	Arbitrary	Arbitrary	Discrete

Why Instantiate [Varian]?

- Validate by comparing with Varian's analytical results for weighted, pay-per-click GSP
  - and obtain computational results on a model of independent interest
- Obtain novel economic results
  - “Apples-to-apples” comparison: how do different auctions perform given identical preferences?
- Most appropriate model is still an open question

Model of Auction Setting

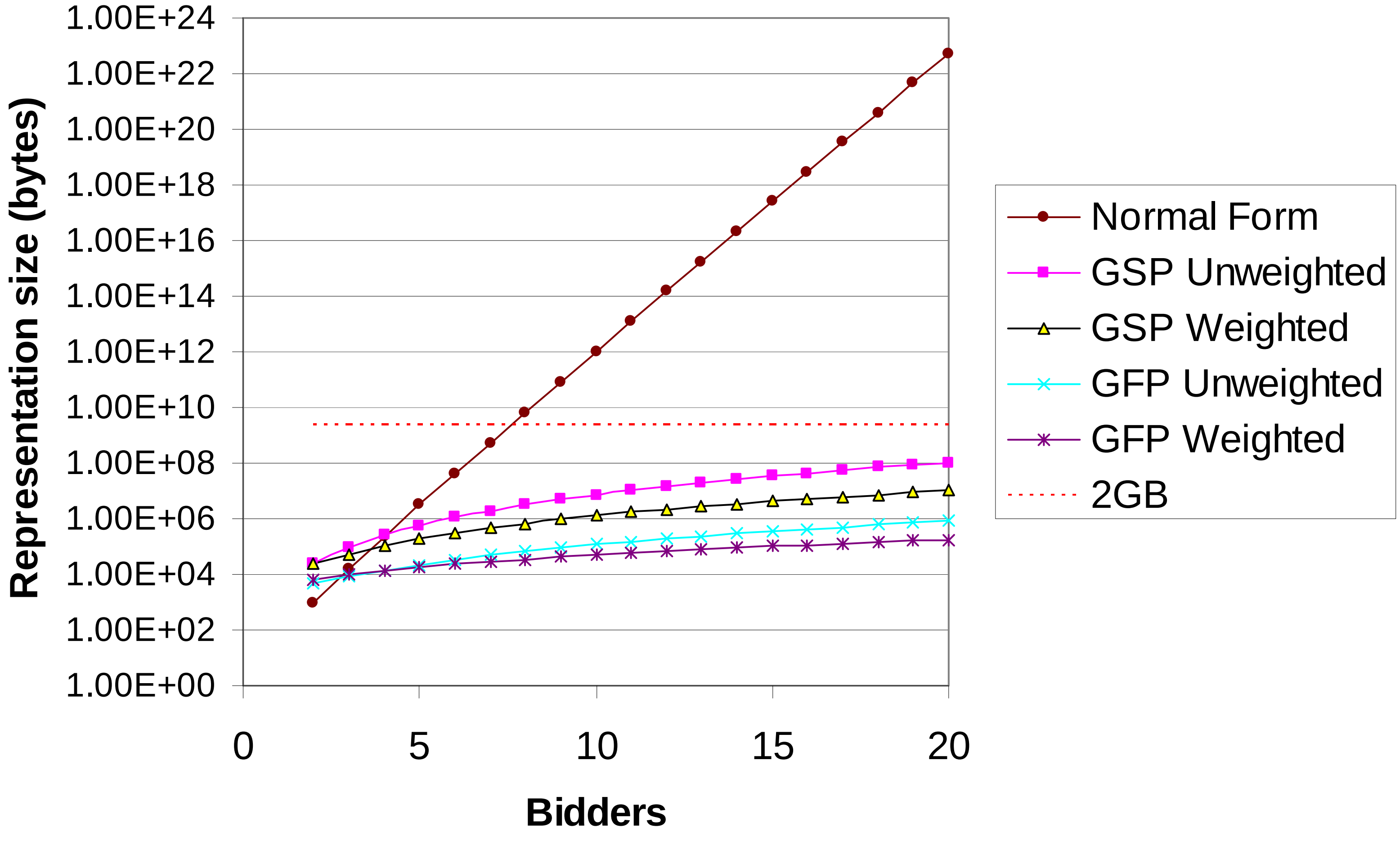
	Weights	CTR across positions	CTR across bidders	Value per Click	Bid Amounts
[EOS]	Always 1	Decreasing	Constant	One value per bidder	Continuous
[Varian]	Arbitrary	Decreasing	Proportional to Weight (“Separable”)	One value per bidder	Continuous
Our model	Arbitrary	Arbitrary	Arbitrary	Arbitrary	Discrete
Problem Distribution	Uniform[0,1]	Uniform[0,1] * CTR of higher slot	Proportional to Weight (“Separable”)	One value per bidder: Uniform[0,1]	Discrete

Experimental Setup

- 10 bidders, 5 slots
- Integer bids between 0 and 10
- For pay-per-click, normalize value/click:
  - Scale max_i $value_i$ to 10, then scale other values proportionately
    - to use full range of discrete bid amounts
- For pay-per-impression, normalize value/impression.

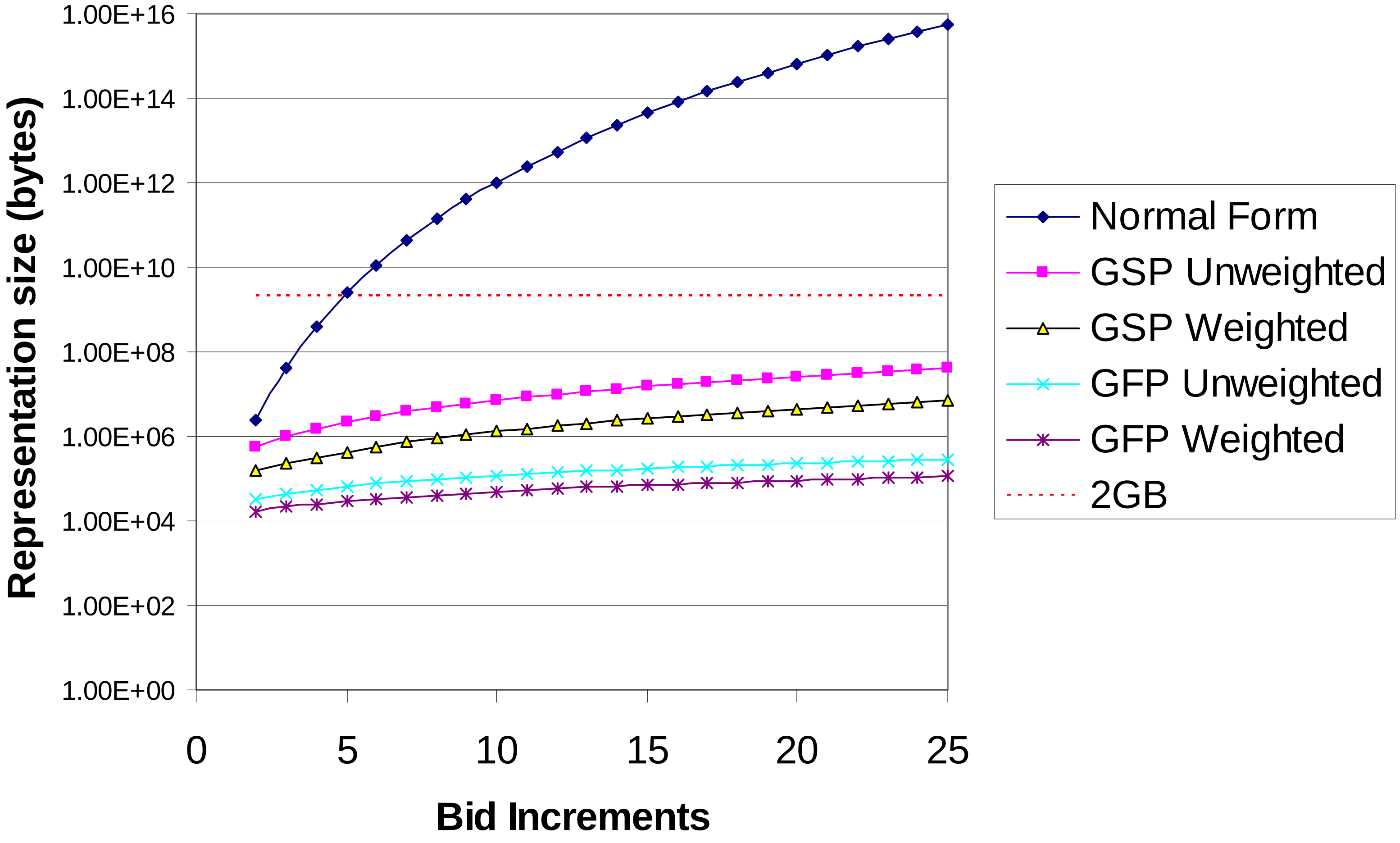
Size Experiments: Players

Integer bids: 0 to 10



Size Experiments: Bid Increments

10 bidders

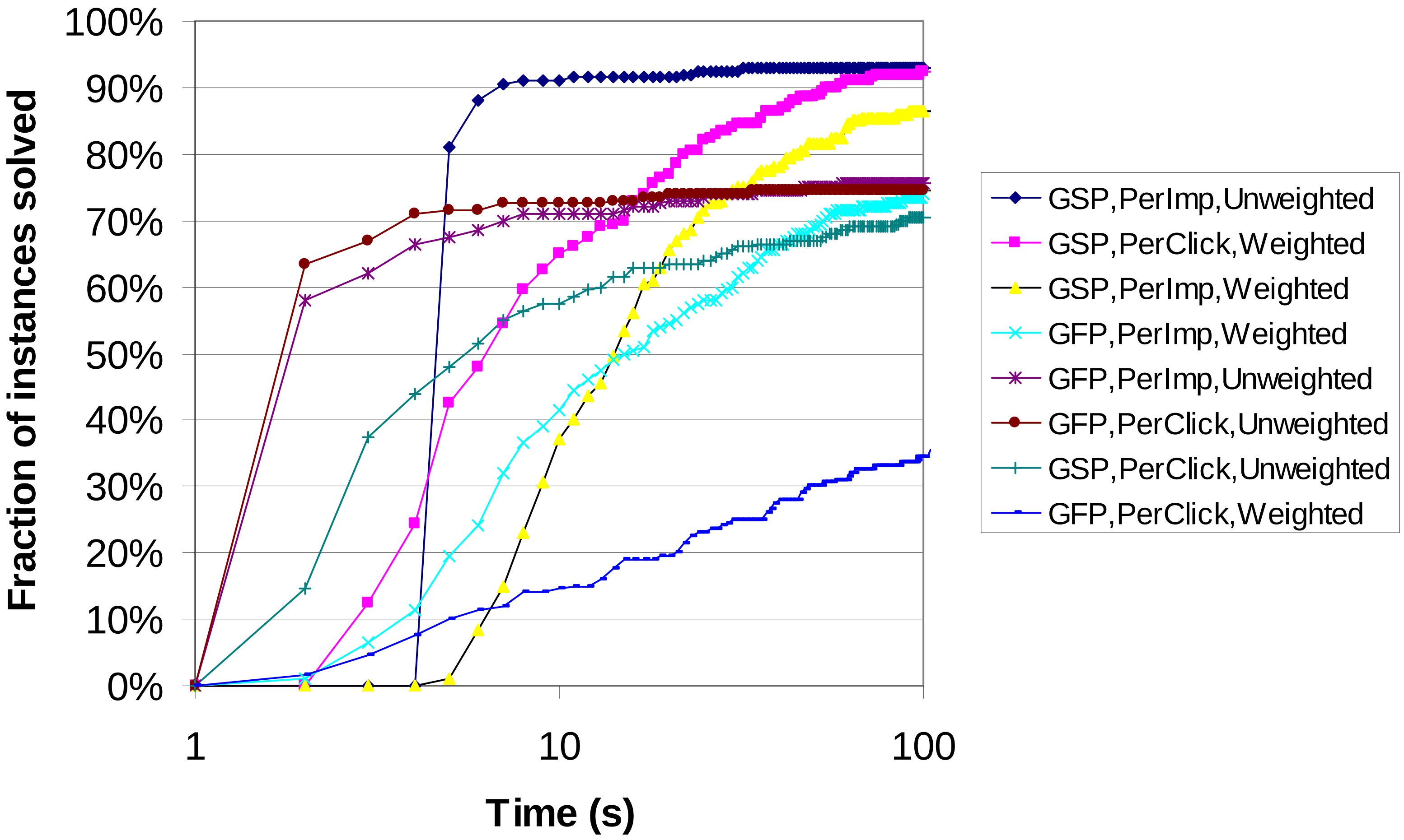


Runtime Experiments: Test-bed

- Environment:
  - Intel Xeon 3.2GHz, 2MB cache, 2GB RAM
  - Suse Linux 10.1
- Solver software:
  - Gambit [McKelvey, McLennan, Turocy, 2007] implementation of simplicial subdivision “simpdiv” [van der Laan, Talman, and van Der Heyden, 1987], AGG-specific dynamic programming inner loop[1] [Jiang & Leyton-Brown, 2006]

1. http://www.cs.ubc.ca/~jiang/agg/

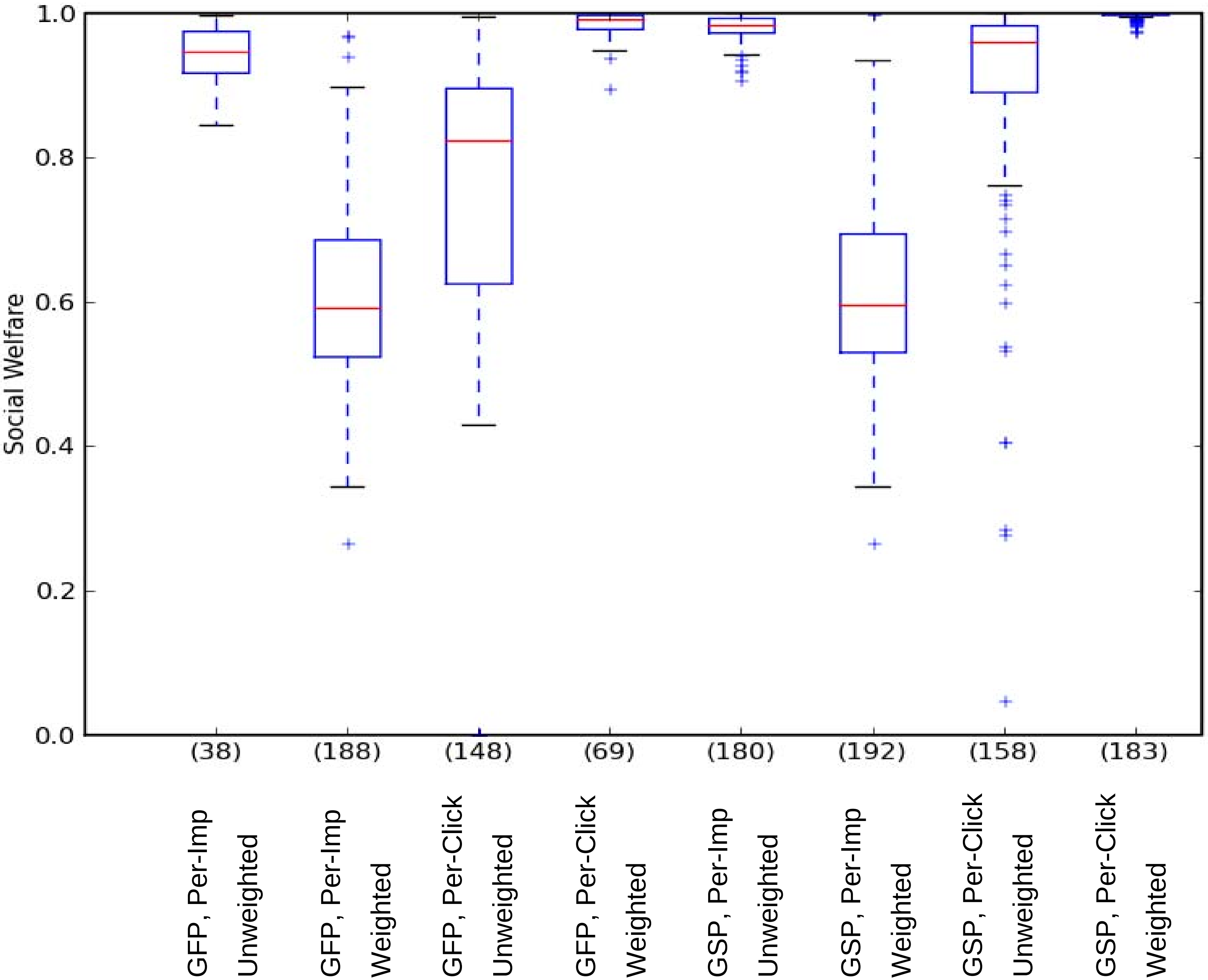
Runtime Experiments: Results



* much longer experiments are ongoing…

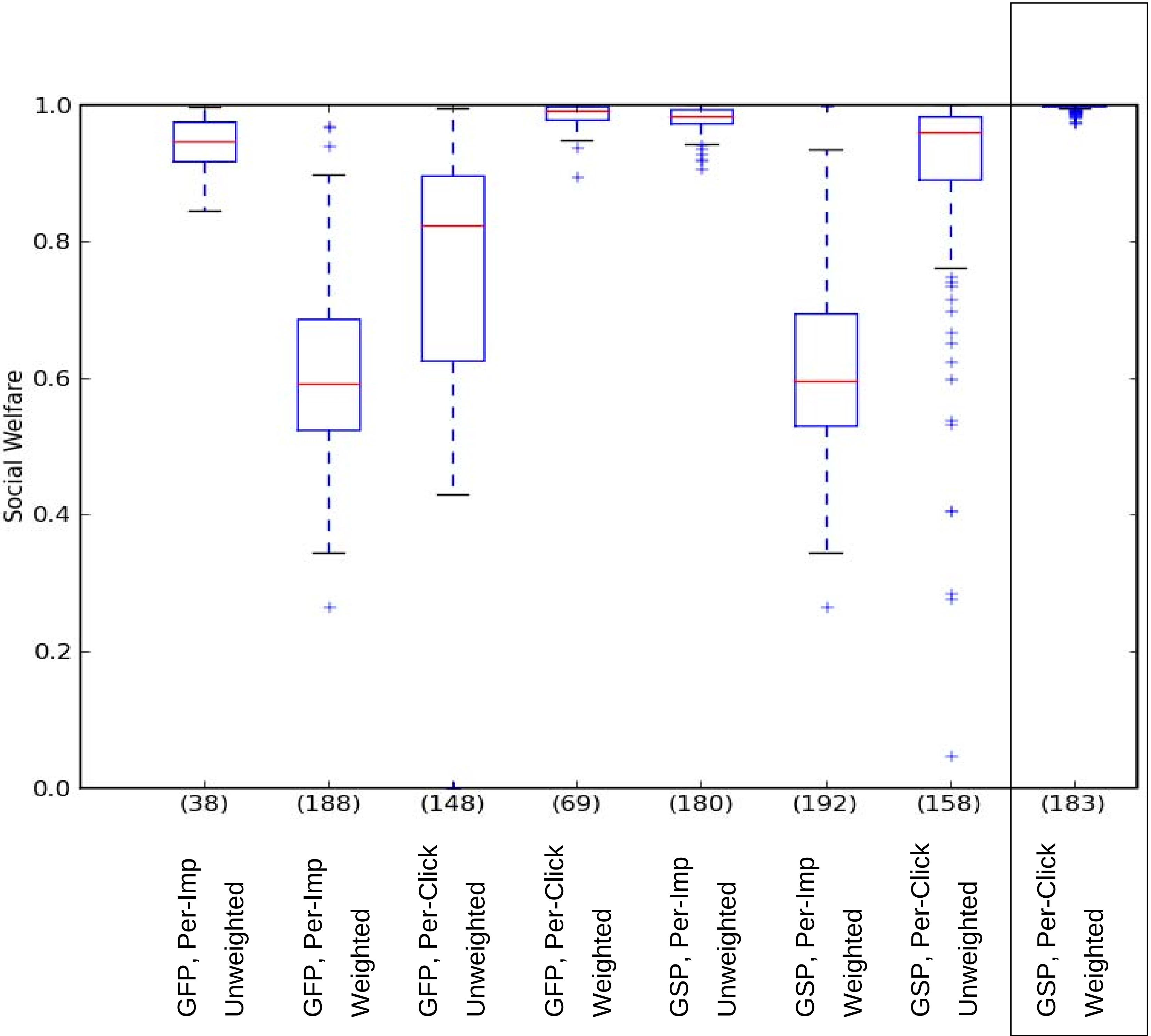
Outline

- Auctions & Model
- Action-Graph Games
- Auctions as AGGs
- Computational Experiments
- Economic Experiments

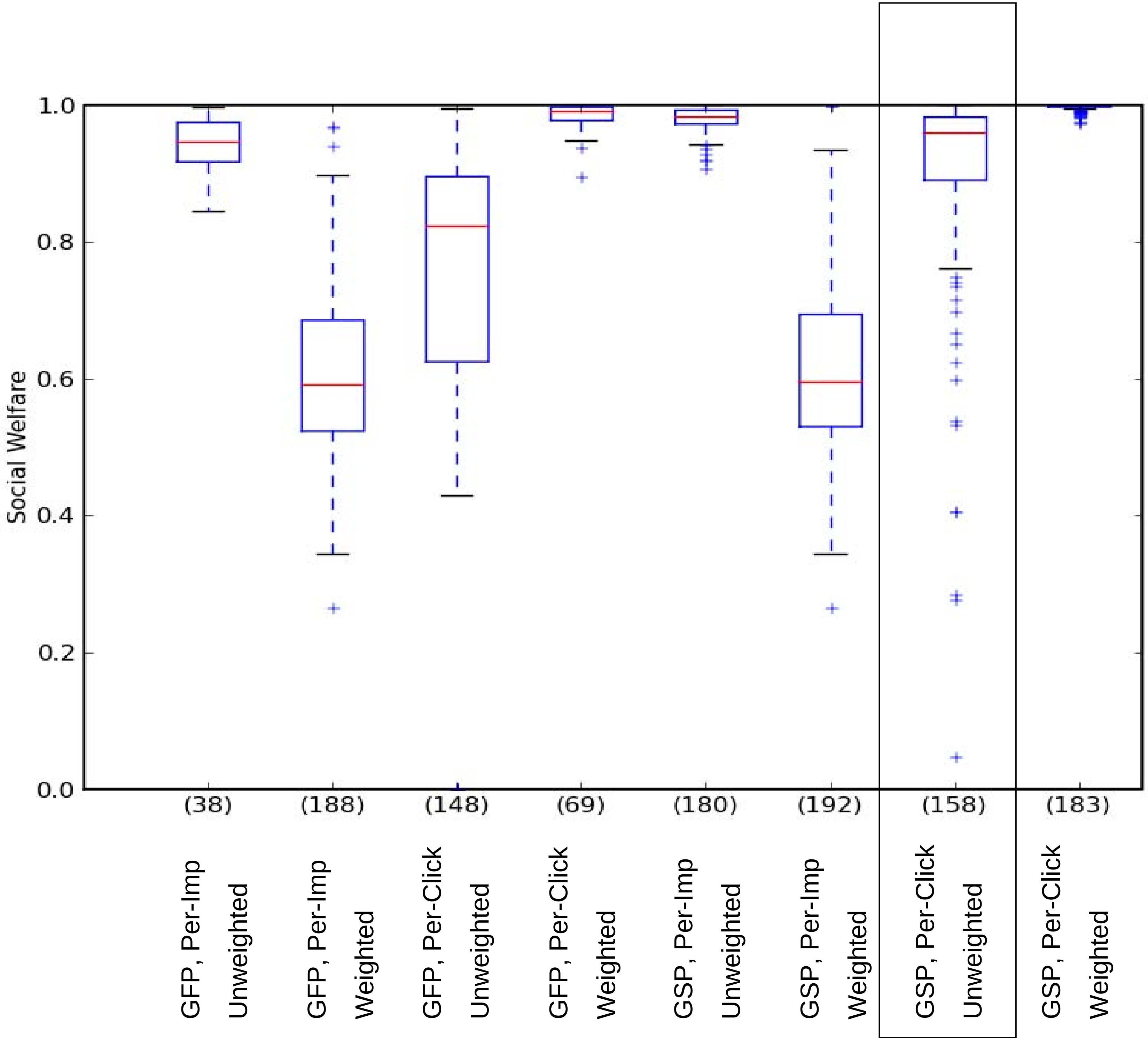


Social Welfare
1.0
0.8
0.6
0.4
0.2
0.0
(38)
(188)
(148)
(69)
(180)
(192)
(158)
(183)
GFP, Per-Imp Unweighted
GFP, Per-Imp Weighted
GFP, Per-Click Unweighted
GFP, Per-Click Weighted
GSP, Per-Imp Unweighted
GSP, Per-Imp Weighted
GSP, Per-Click Unweighted
GSP, Per-Click Weighted

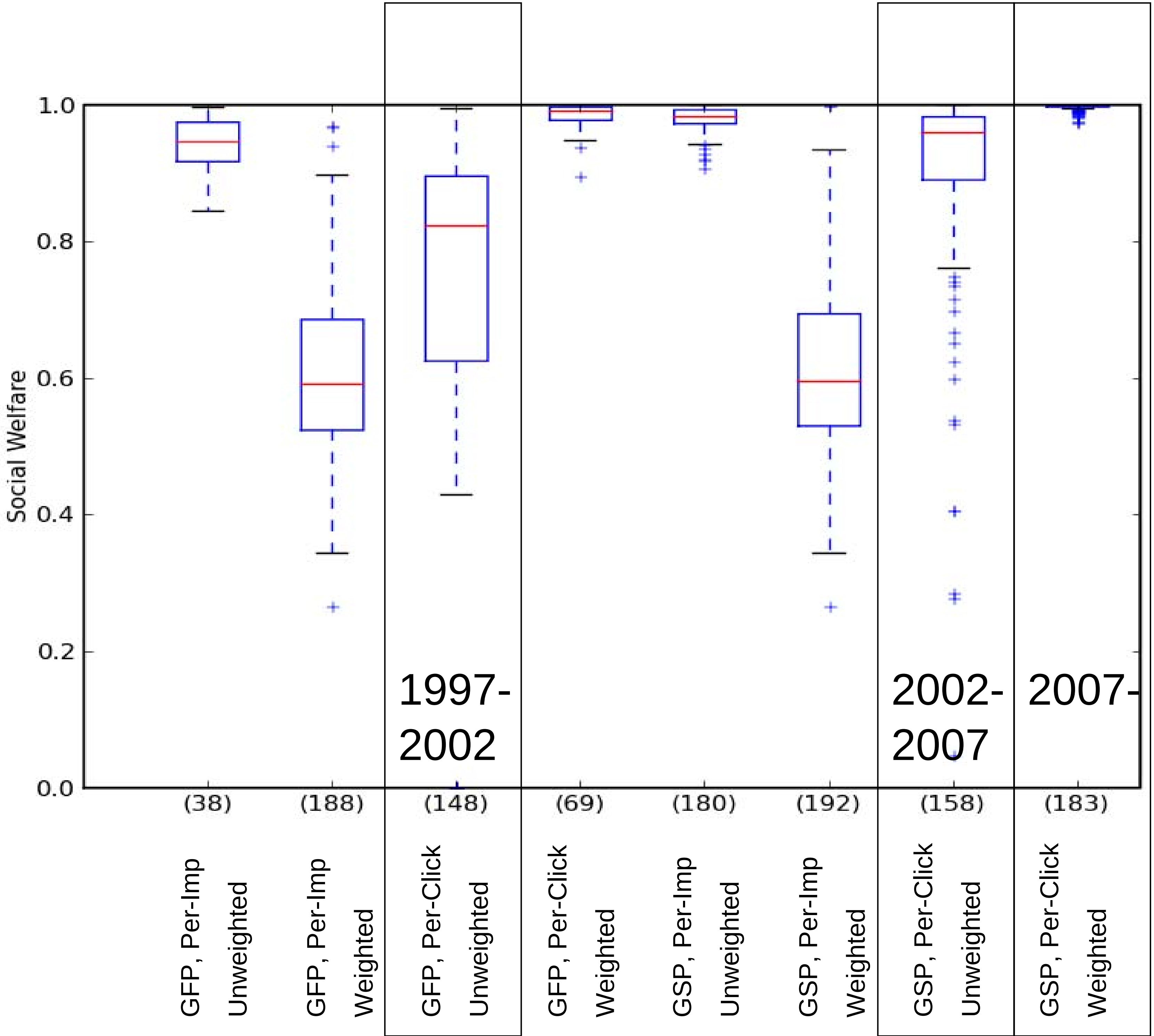
[Varian]: Any SNE gives rise to an efficient allocation



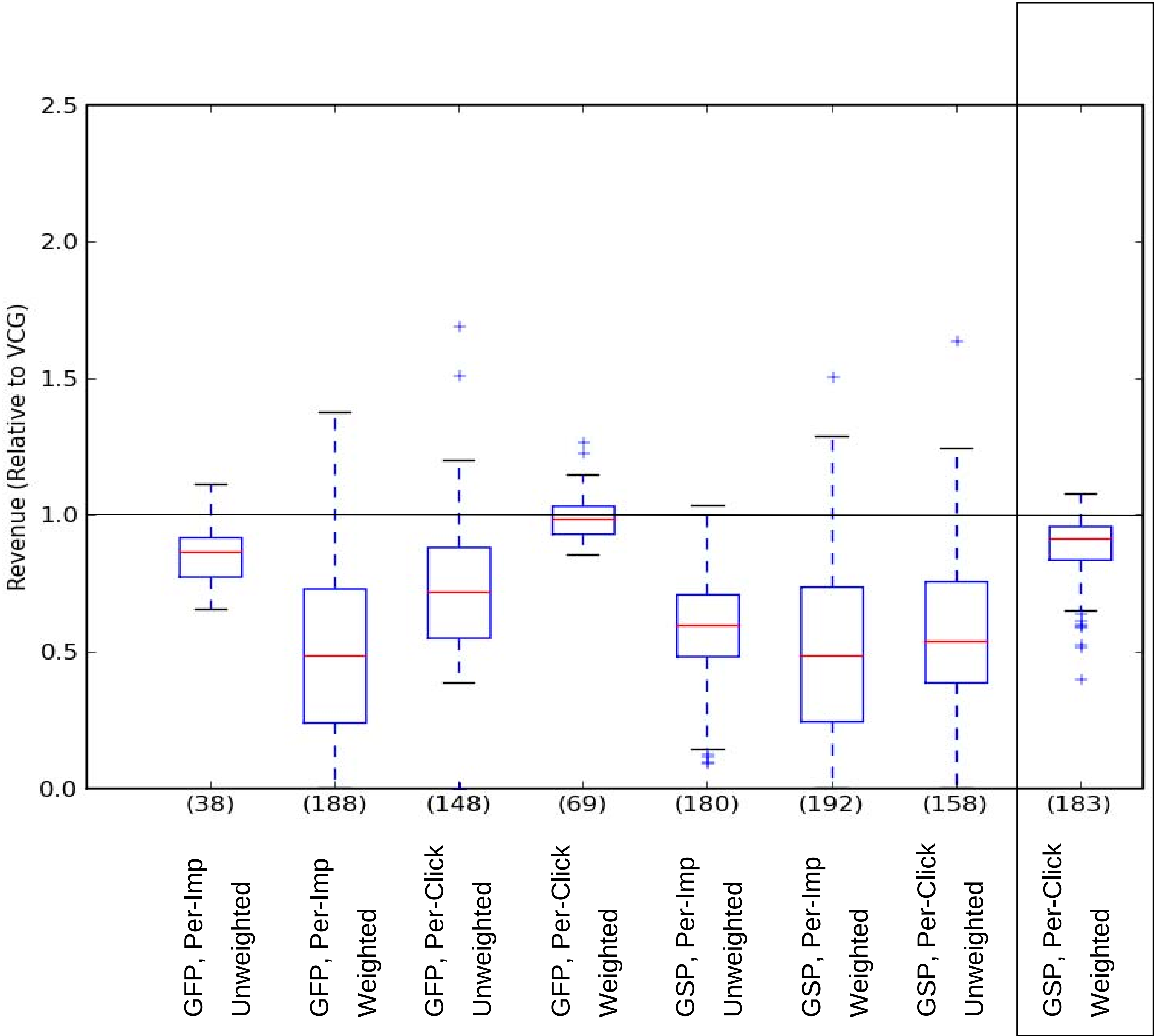
[EOS]'s auction with [Varian]'s preference model



Yahoo! Auctions: Past and Present

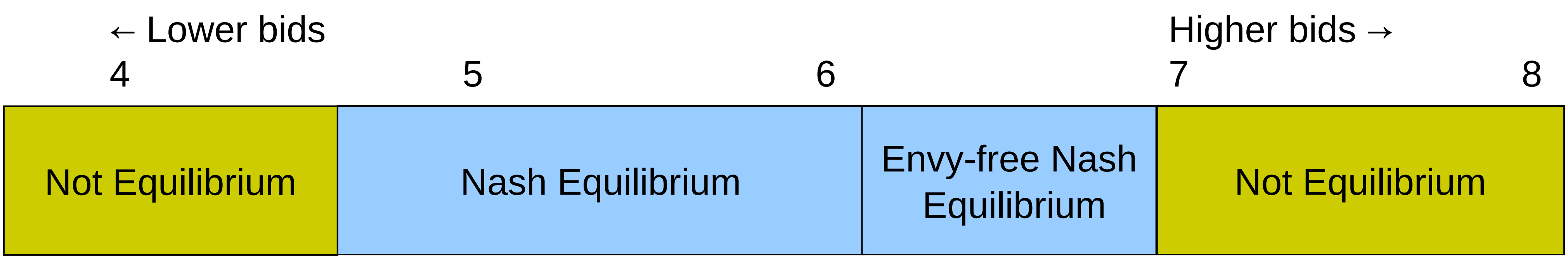


[Varian]: VCG revenue is a lower bound on SNE revenue



Multiple Equilibria of GSPs [Varian; EOS]

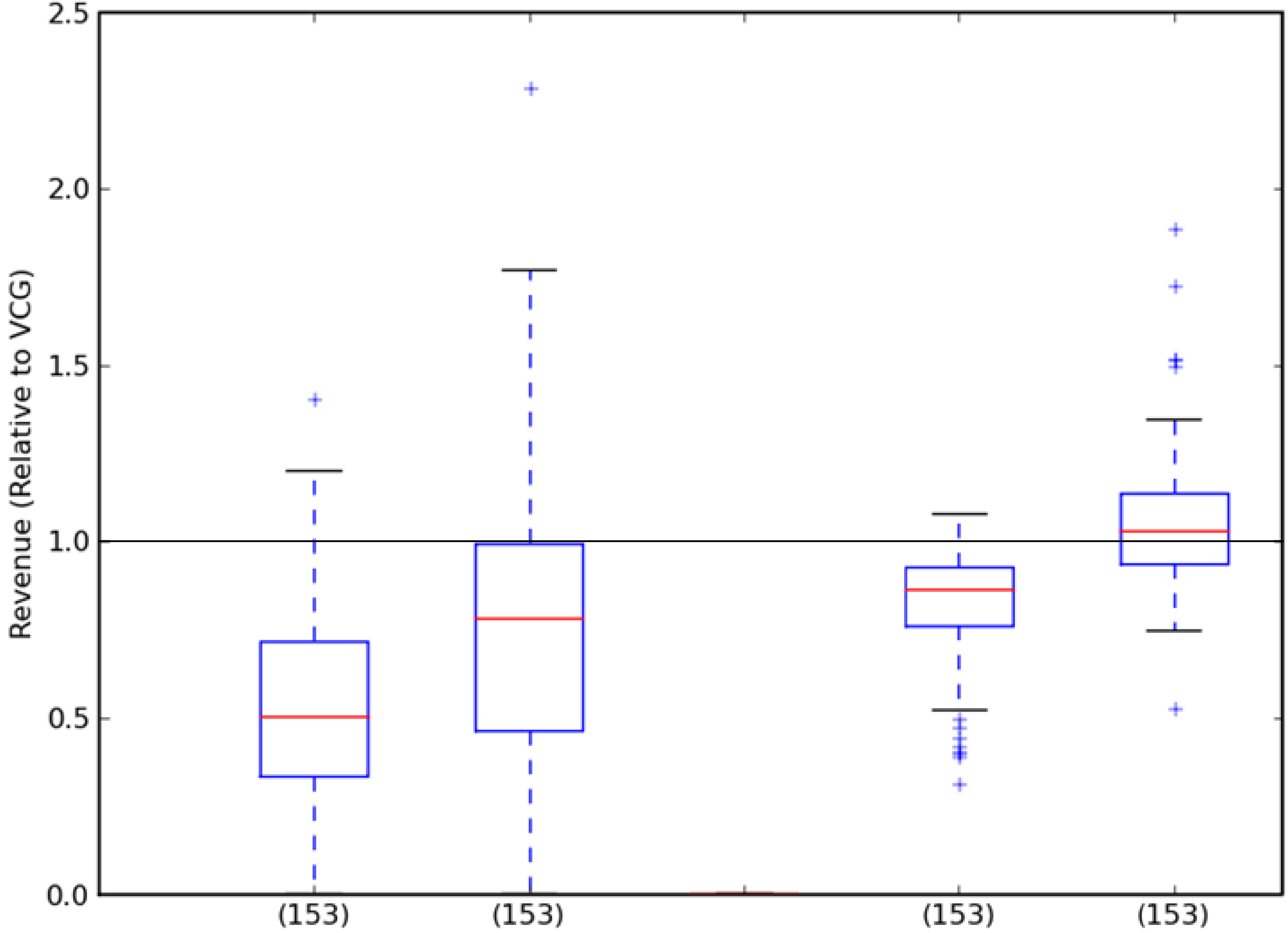
- Each agent can have many best responses to an equilibrium strategy profile.
  - Raising i's bid increases (i-1)'s price, decreasing i's envy.
- Given an envy-free NE / SNE, lowering an agent’s bid may lead to an efficient, pure NE w/ sub-VCG revenue



- Even if pure NE exist for continuous bids, they may not exist for discrete bids.

Equilibrium selection

- Previous results simply showed the first equilibrium found by simpdiv
  - Often a mixed strategy over arbitrary points on equilibrium interval
- Local search approach to equilibrium selection:
  - Start point: Nash equilibrium found by simpdiv
  - Neighbours: *Nash equilibria* where one bid is changed by one increment
  - Objective: maximize/minimize sum of bids
  - Algorithm: Greedily raise bids (choose bidder by random permutation); random restarts.



Yahoo!

2002-2007

Google /

Yahoo! 2007-

Summary

- Many position auctions are tractable:
  - Polynomial-size AGG
  - Polynomial-time expected utility by dynamic programming
- Very general preference model:
  - Position-specific valuations
  - Non-separable CTRs (and arbitrary weights)
- Experimental results consistent with existing theory and practice.

Future Work

- Economic:
  - Use full preference model (learn from data)
  - Model richer preferences (e.g. cascading CTR [Aggarwal, et al, 2008; Kempe, Mahdian, 2008])
- Computational:
  - *In progress:* Adapt SEM [Porter, Nudelman, Shoham, 2006] to AGGs: Allows enumerating equilibria (answer questions like “what percentage of pure equilibria are envy free?”)

Thank You.