EconGames.org

Andrei Dubovik Clemens Fiedler Vladimir Karamychev
Bart Voogt Gijsbert Zwart

April 16, 2018

Talk Outline

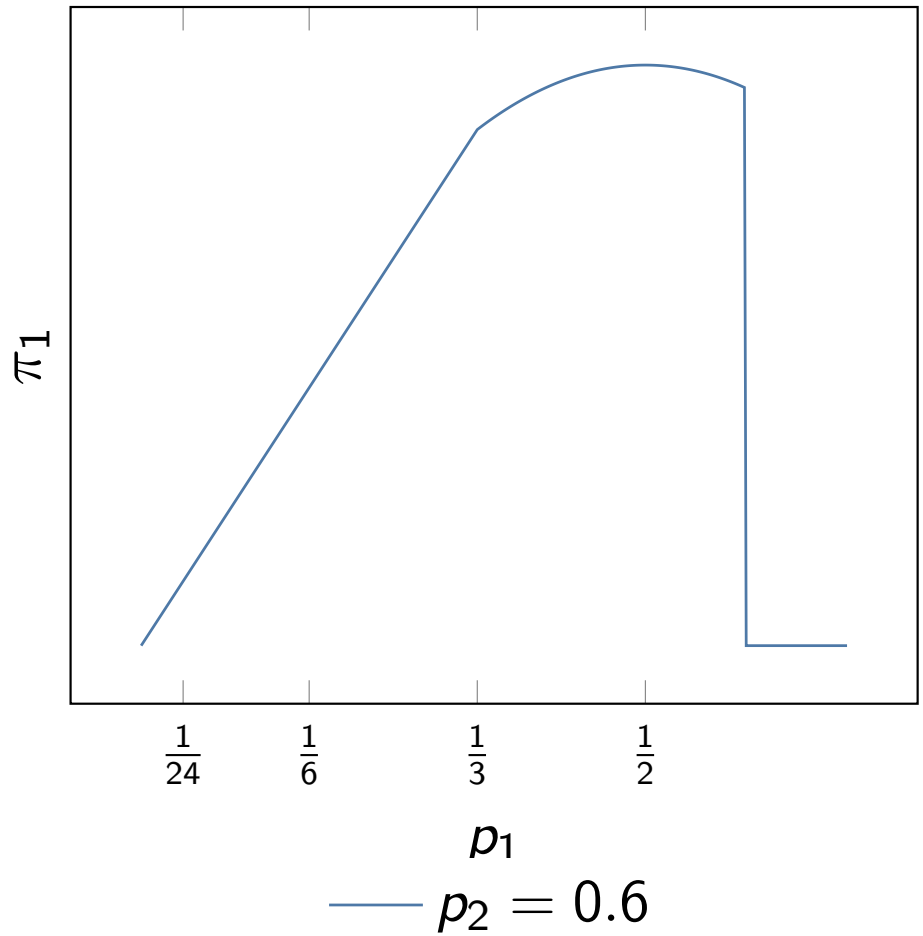
1. First game: Bertrand-Edgeworth Duopoly
2. Speed tournament
3. Analysis of the Erasmus tournament
4. Setting up `https://econgames.org`
5. Setting up Python framework
6. Motivation and long-term goals
7. Next tournament

Bertrand-Edgeworth Duopoly

- Levitan and Shubik (1972)
- Two firms choose prices p
- Capacity constraint $k = \frac{2}{3}$
- Efficient rationing
- Suppose $p_1 < p_2$, then

$$q_1 = \min(1 - p_1, k)$$
$$q_2 = \min(1 - q_1 - p_2, k)$$

- Simultaneous move game
- Mixed Nash over continuous support

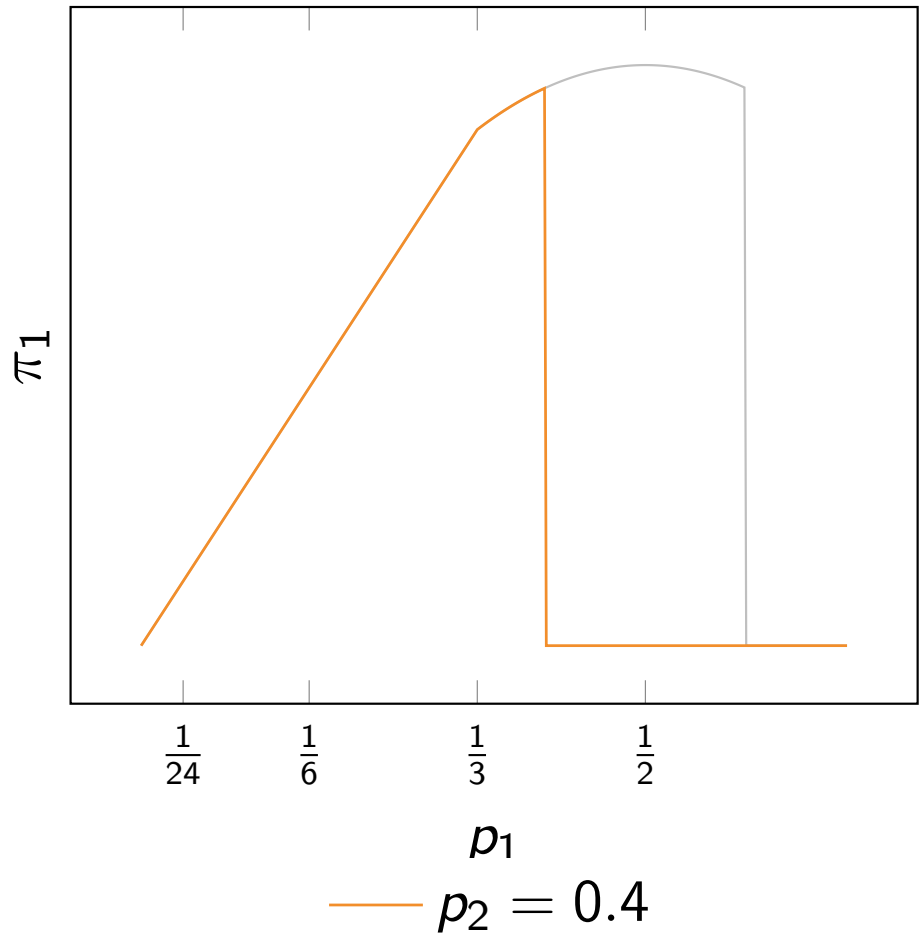


Bertrand-Edgeworth Duopoly

- Levitan and Shubik (1972)
- Two firms choose prices p
- Capacity constraint $k = \frac{2}{3}$
- Efficient rationing
- Suppose $p_1 < p_2$, then

$$q_1 = \min(1 - p_1, k)$$
$$q_2 = \min(1 - q_1 - p_2, k)$$

- Simultaneous move game
- Mixed Nash over continuous support

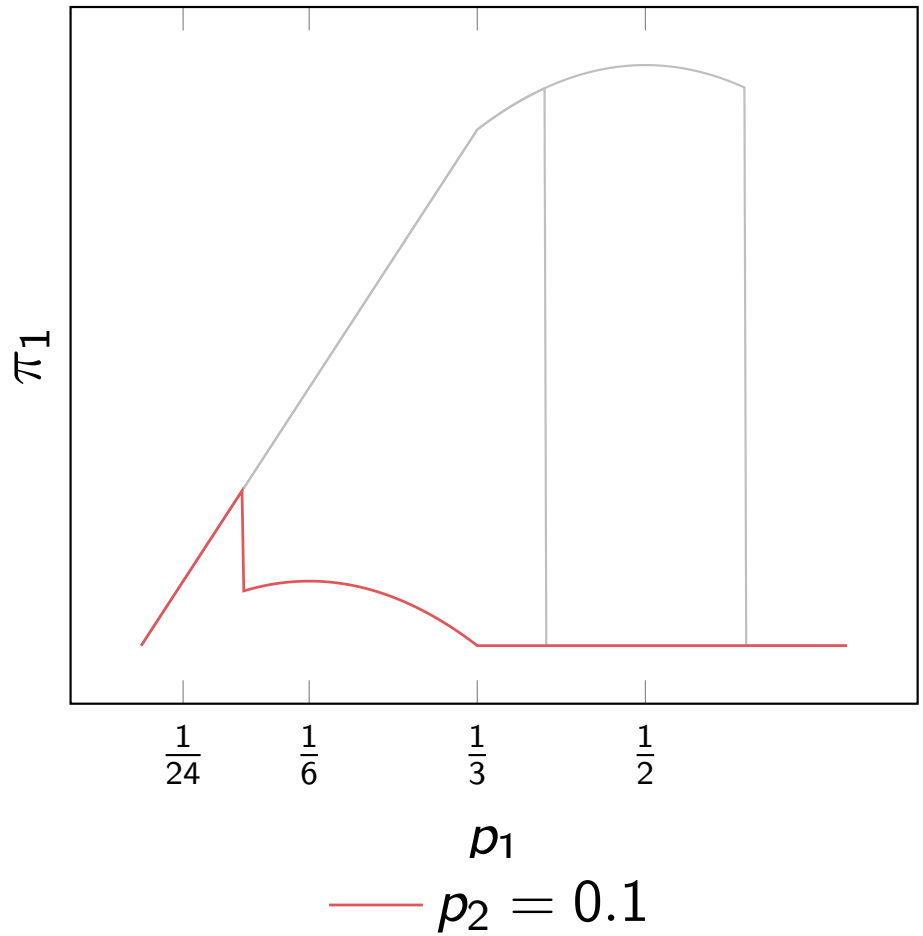


Bertrand-Edgeworth Duopoly

- Levitan and Shubik (1972)
- Two firms choose prices p
- Capacity constraint $k = \frac{2}{3}$
- Efficient rationing
- Suppose $p_1 < p_2$, then

$$q_1 = \min(1 - p_1, k)$$
$$q_2 = \min(1 - q_1 - p_2, k)$$

- Simultaneous move game
- Mixed Nash over continuous support

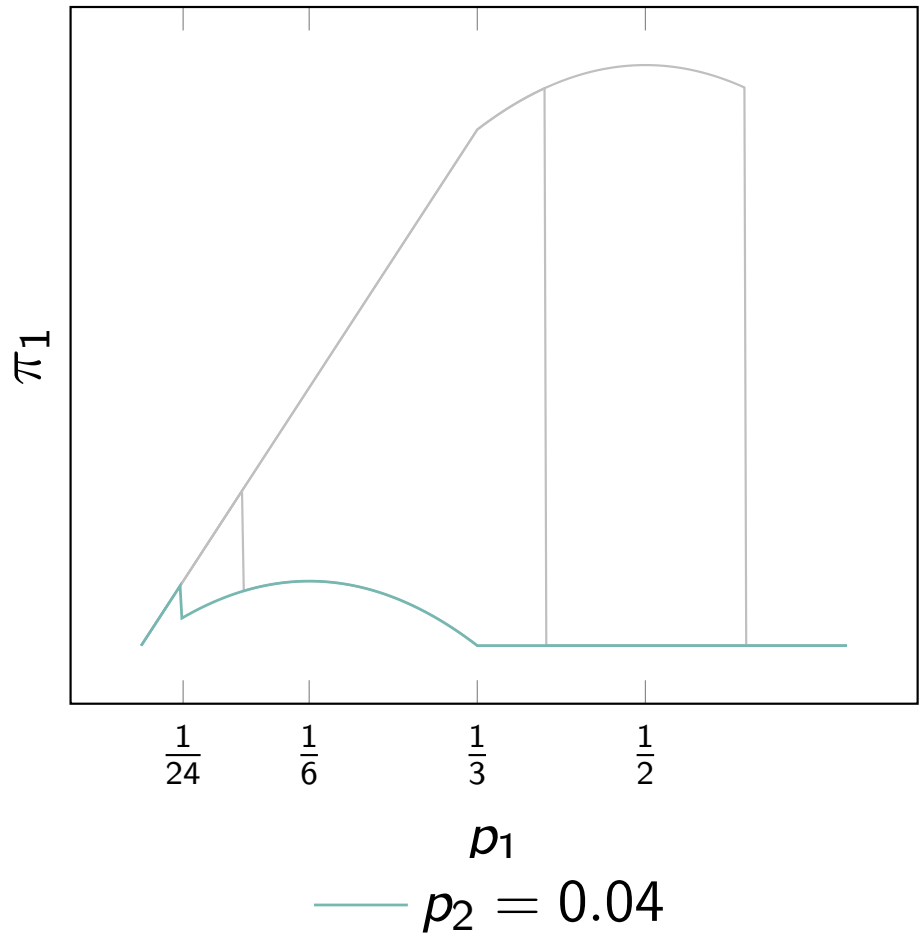


Bertrand-Edgeworth Duopoly

- Levitan and Shubik (1972)
- Two firms choose prices p
- Capacity constraint $k = \frac{2}{3}$
- Efficient rationing
- Suppose $p_1 < p_2$, then

$$q_1 = \min(1 - p_1, k)$$
$$q_2 = \min(1 - q_1 - p_2, k)$$

- Simultaneous move game
- Mixed Nash over continuous support



Speed Tournament

General organization

- N players
- $\frac{N^2-N}{2}$ round-robin matches
- Each match consists of 1000 periods
- Match history is available
- Cross-match history is not available
- The player with most total profits wins

Speed tournament

- 15 minutes to program a strategy
- 2 person groups are allowed

Erasmus Tournament (8 September 2017)

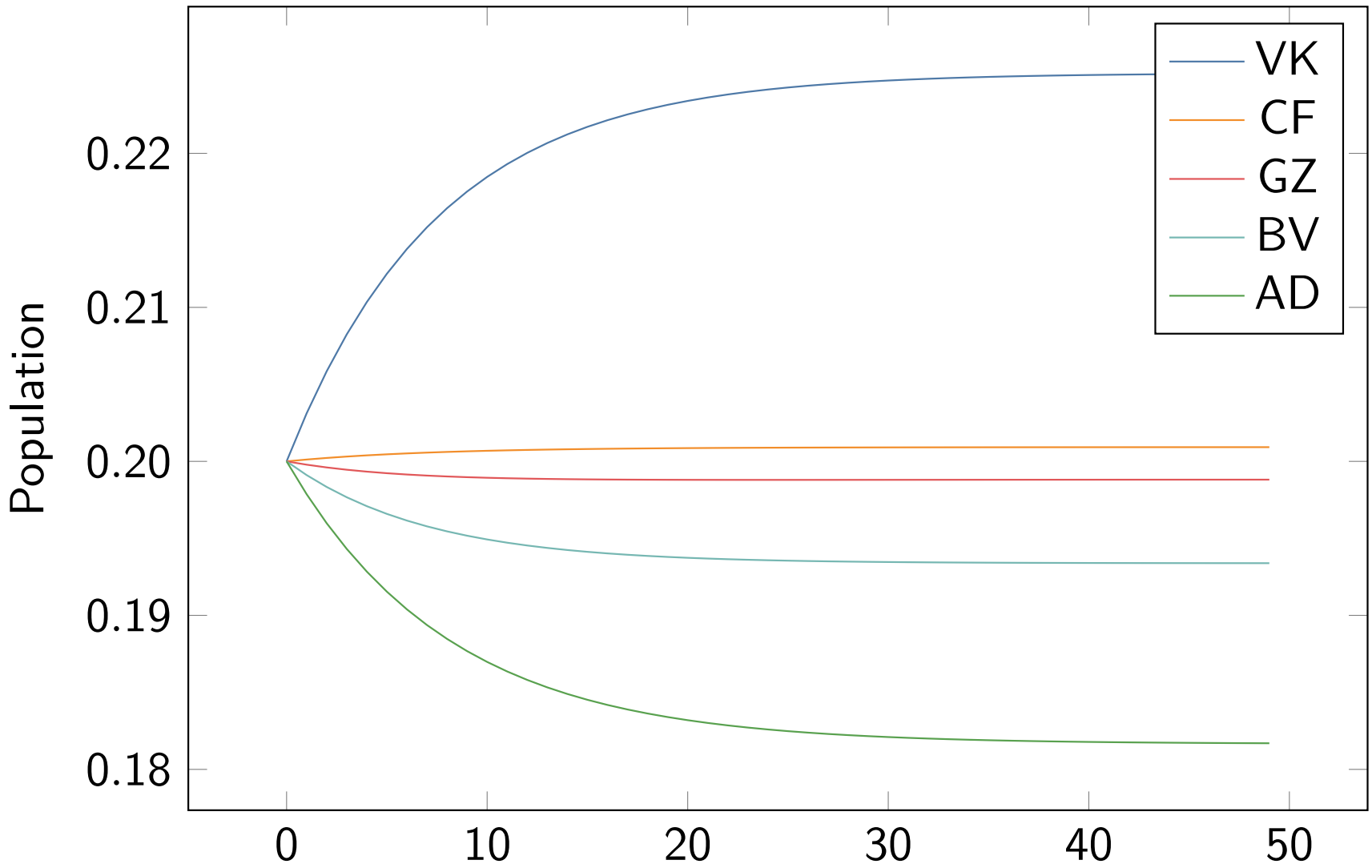
Profit Matrix

	VK	CF	GZ	BV	AD
VK		32.2, 27.1	56.1, 32.2	27.2, 32.0	25.1, 23.4
CF	27.1, 32.2		46.6, 33.6	23.7, 22.4	25.0, 23.1
GZ	32.2, 56.1	33.6, 46.6		27.2, 36.6	27.3, 37.6
BV	32.0, 27.2	22.4, 23.7	36.6, 27.2		25.2, 24.6
AD	23.4, 25.1	23.1, 25.0	37.6, 27.3	24.6, 25.2	
Total	140.6	122.3	120.3	116.2	108.7

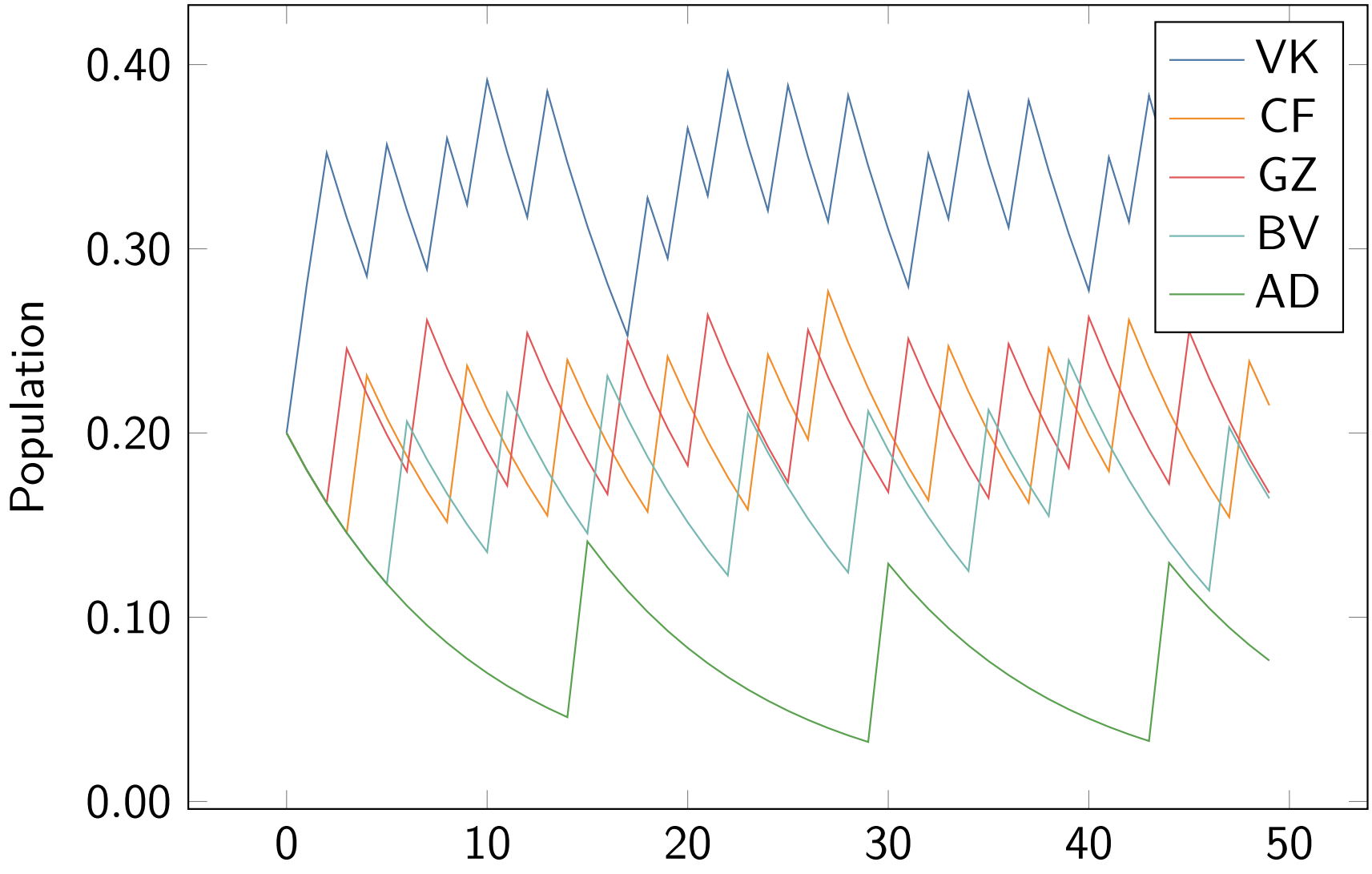
Comments:

- VK played best responses to kernel density estimations
- BV and AD played static strategies
- GZ lost every round robin but took 3rd place overall

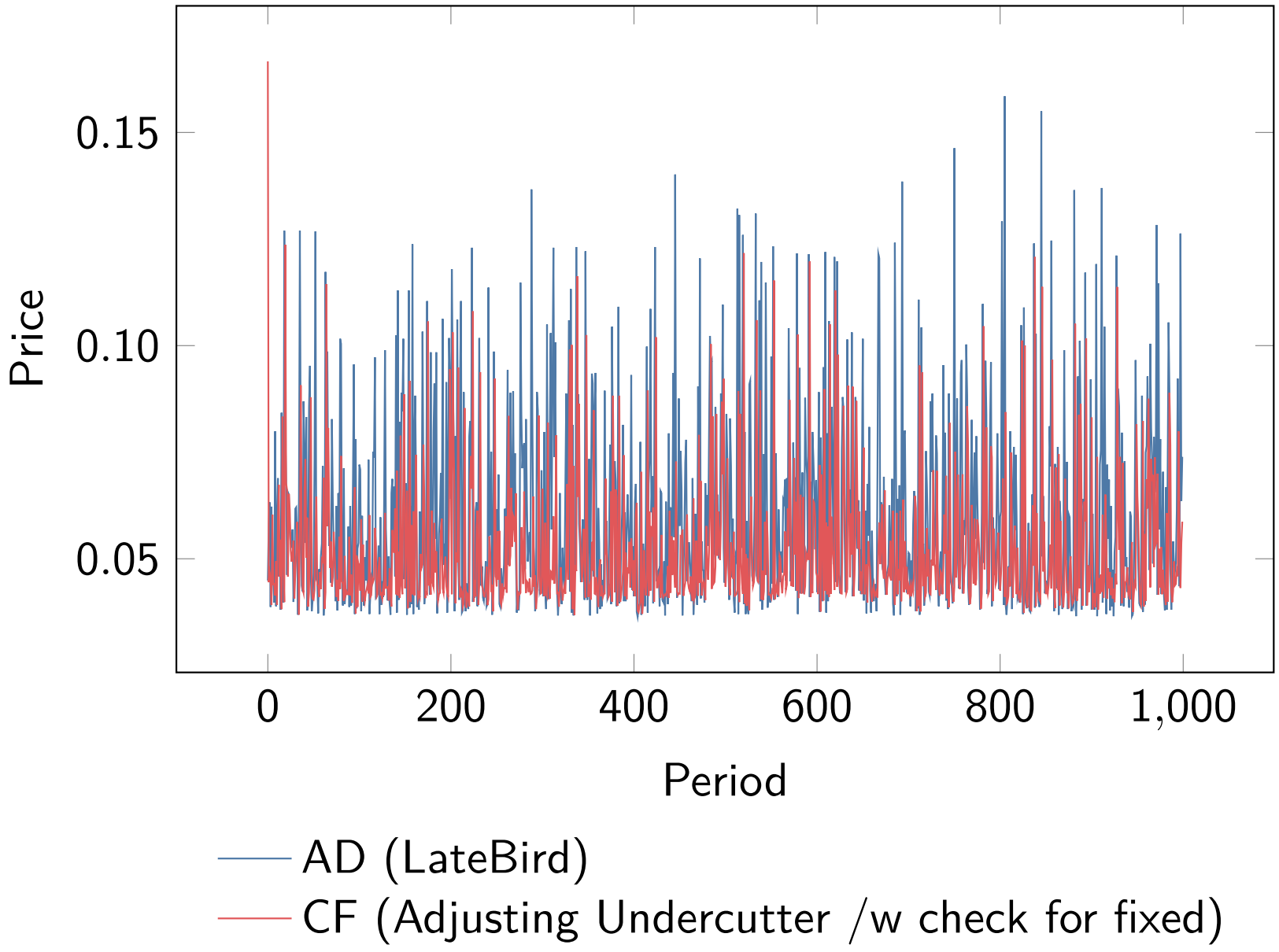
Evolutionary Dynamics (Proportional)



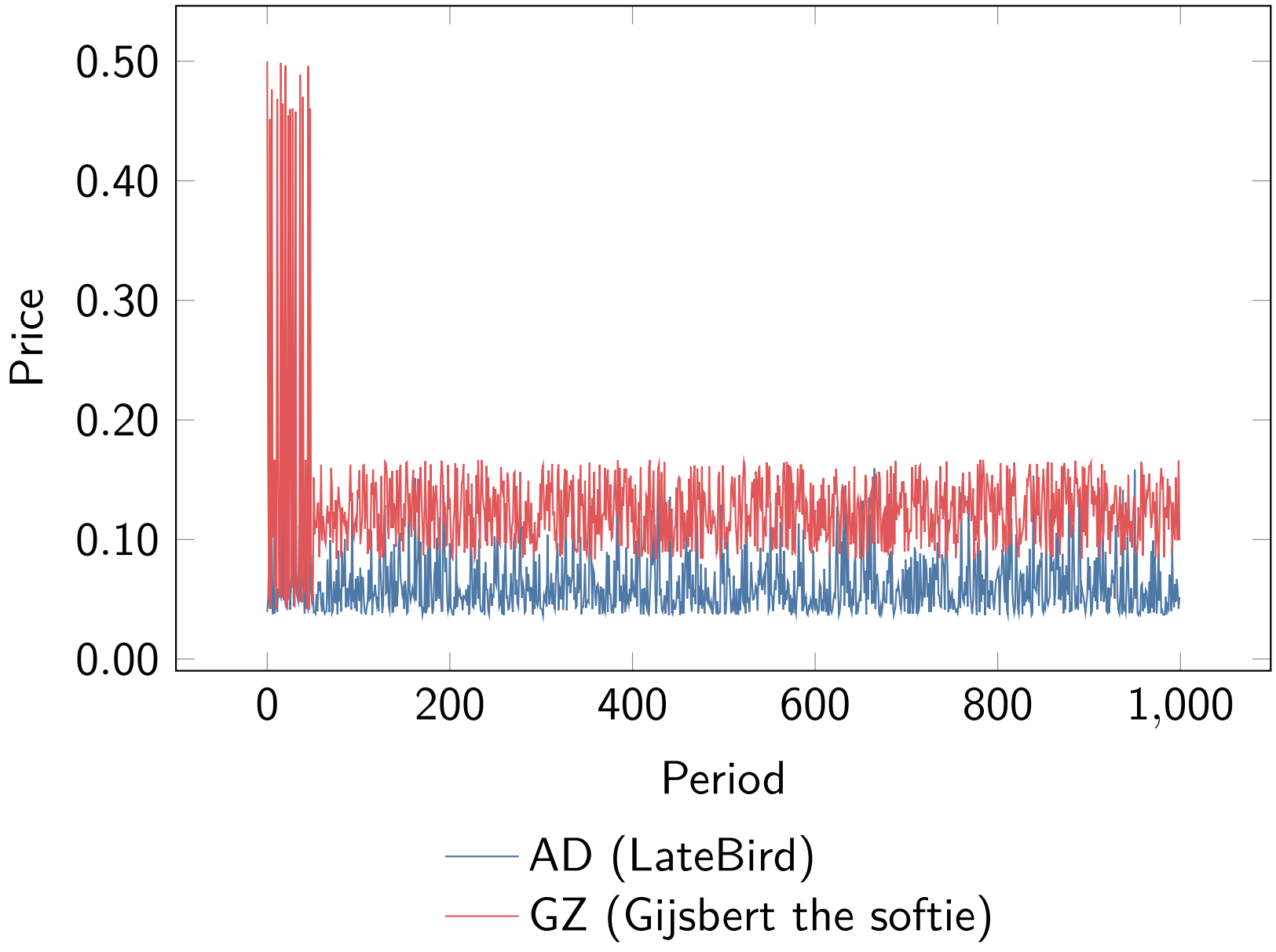
Evolutionary Dynamics (Winner Takes It All)



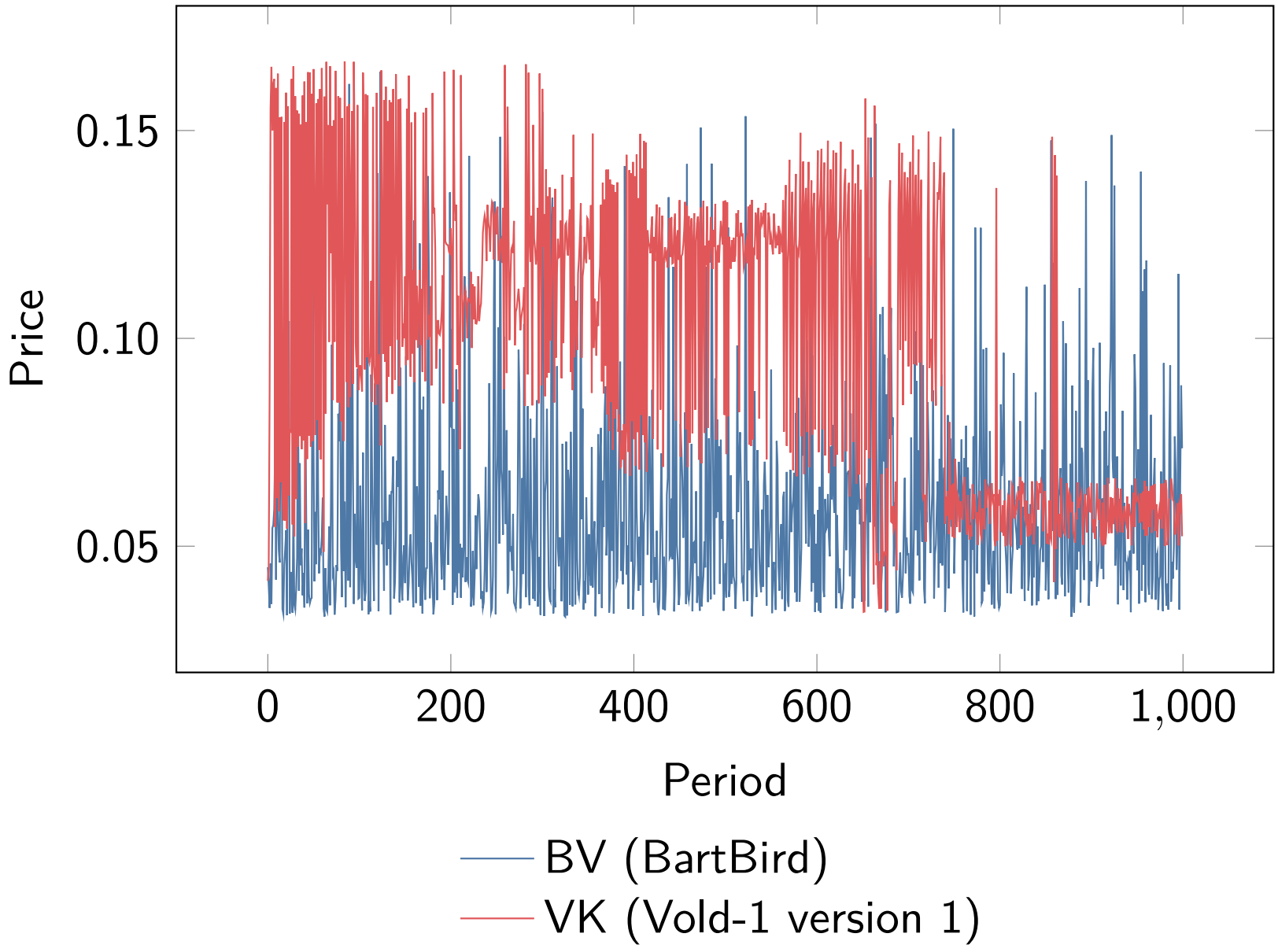
Price Dynamics



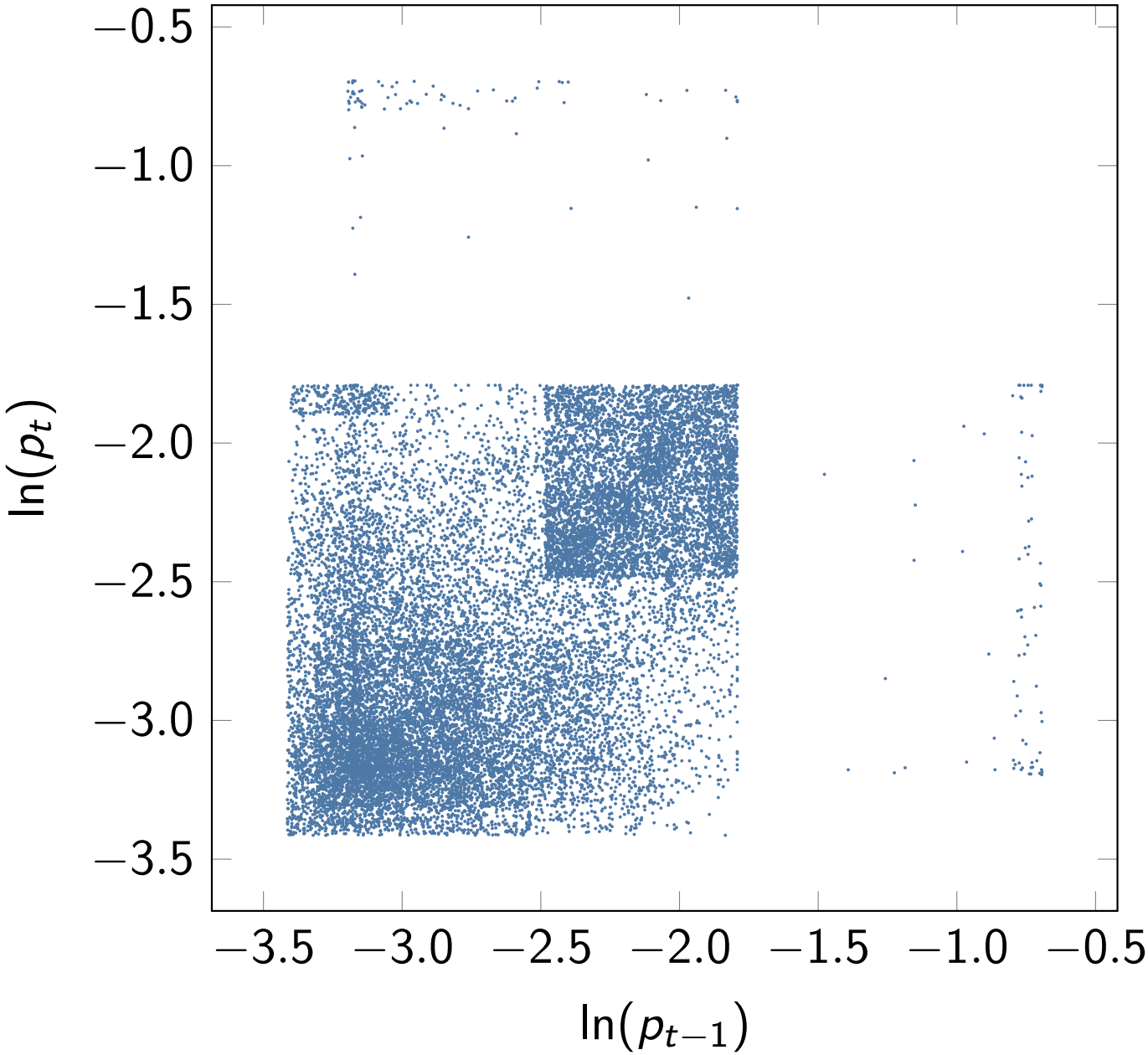
Price Dynamics



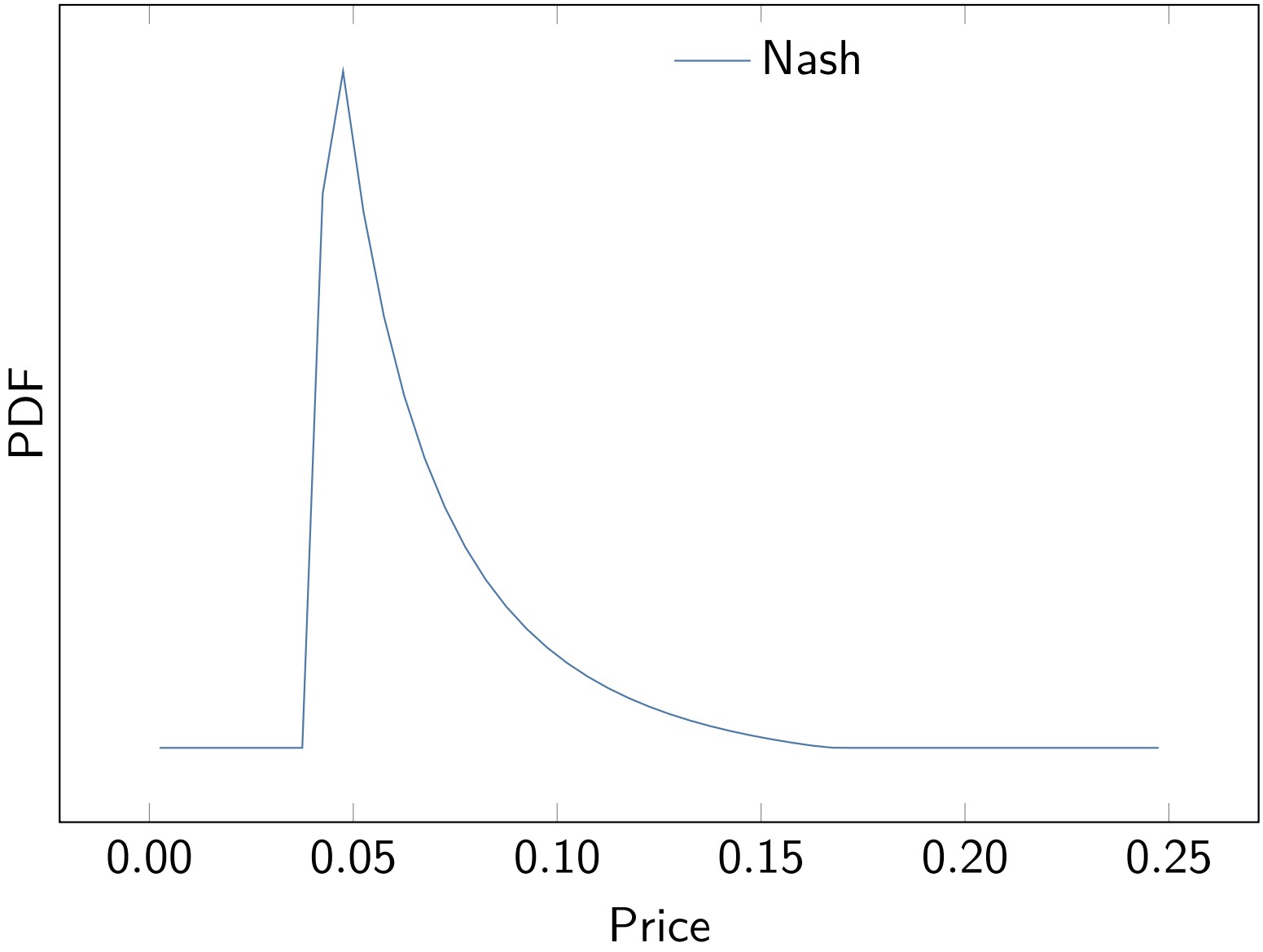
Price Dynamics



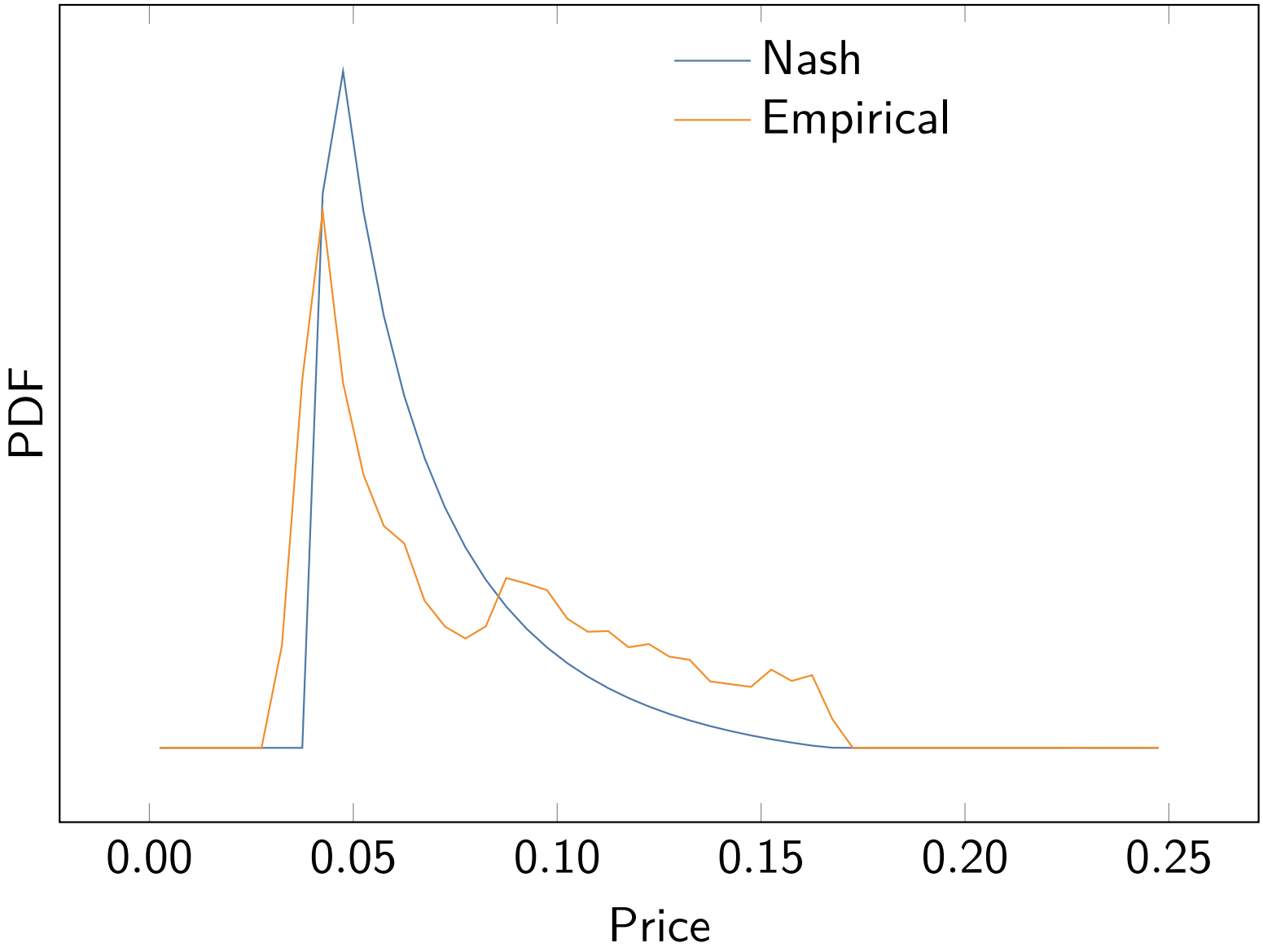
Autocorrelation



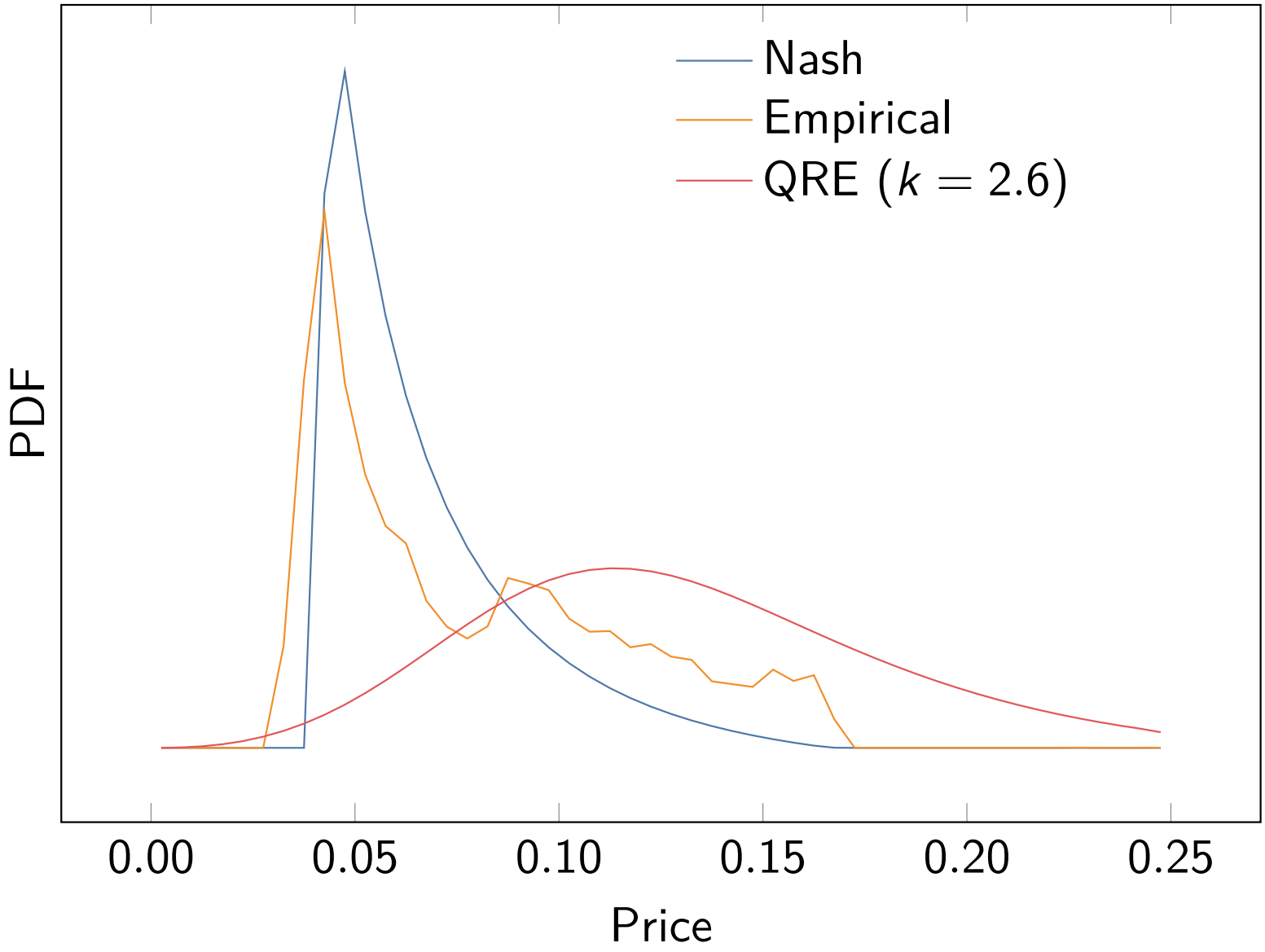
Price Distributions



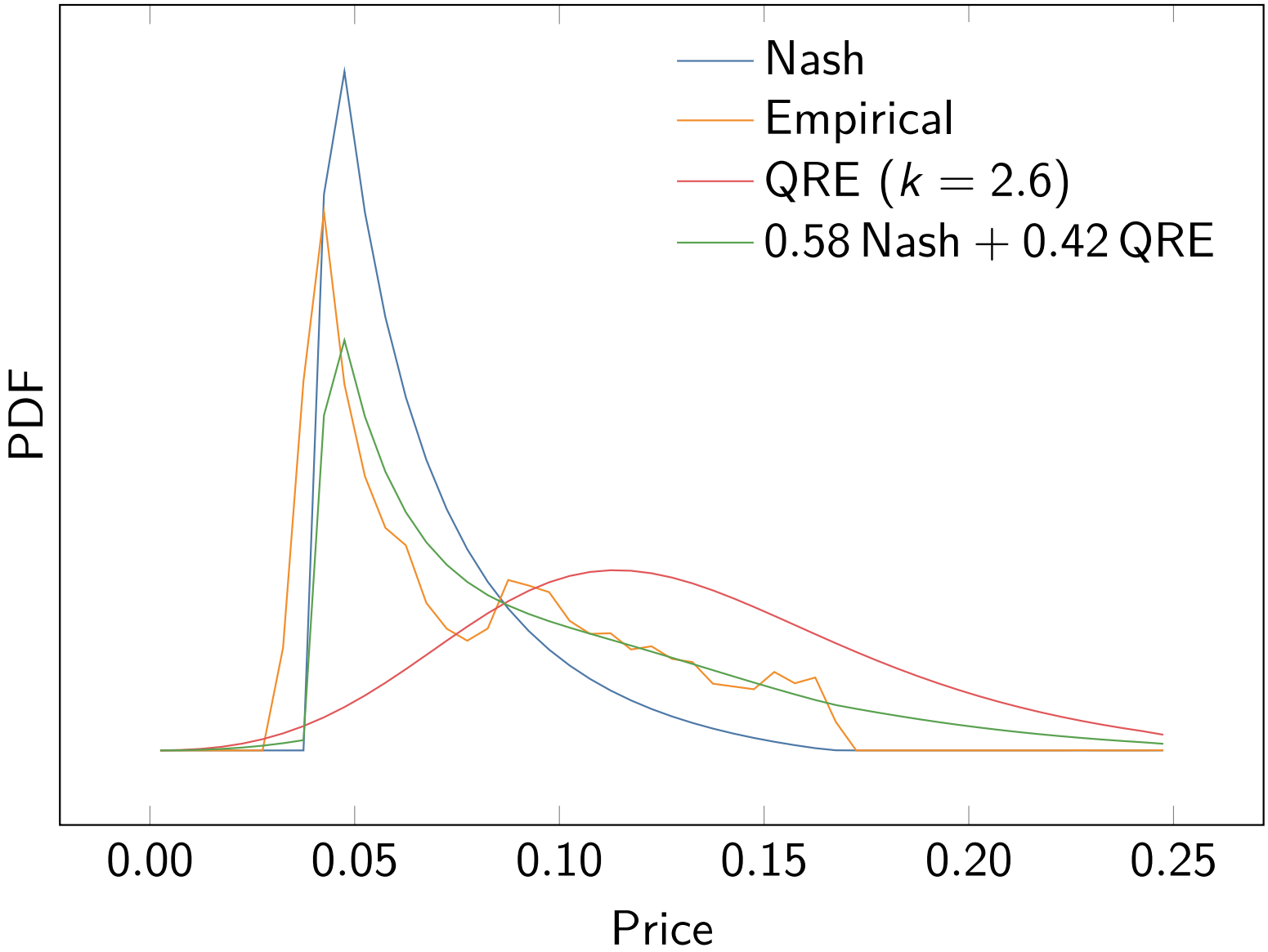
Price Distributions



Price Distributions

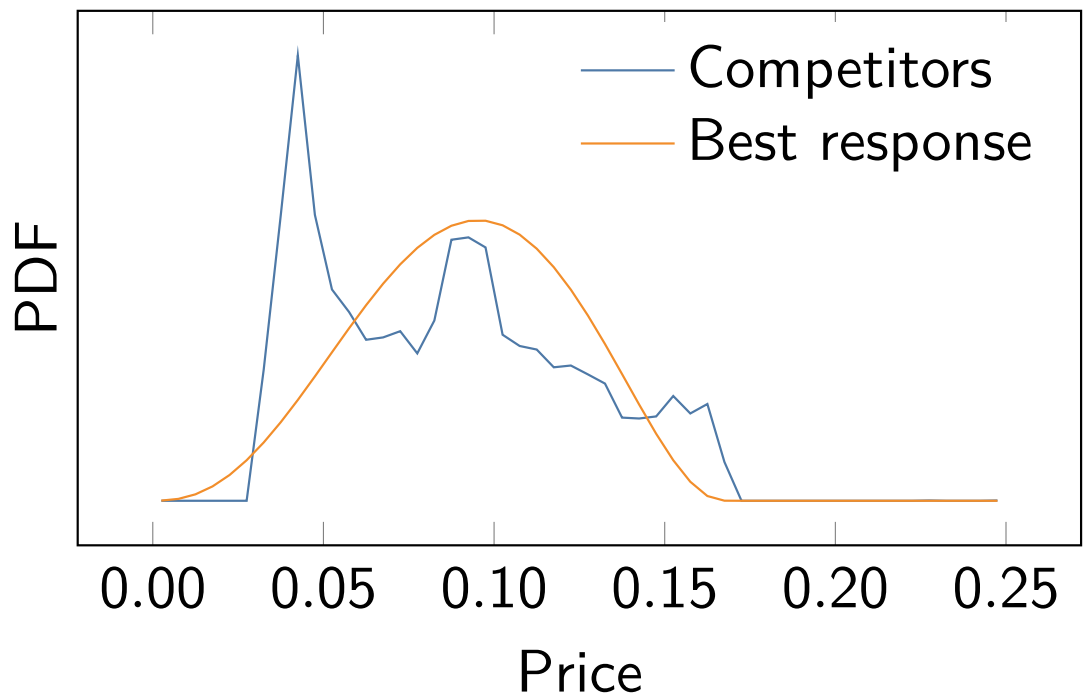


Price Distributions



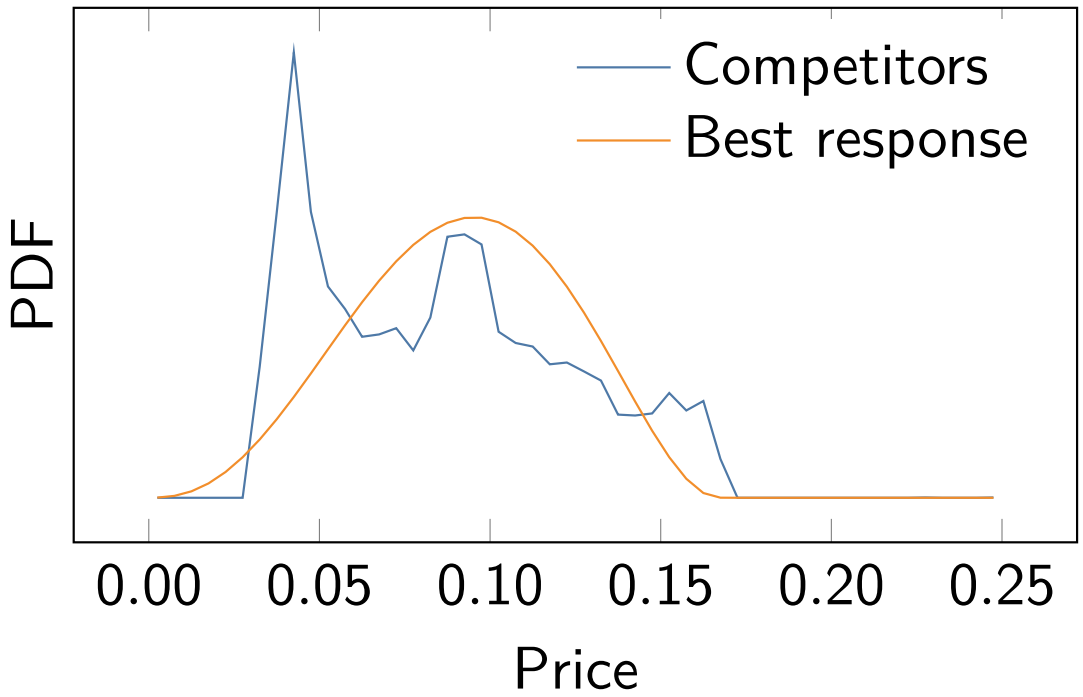
Optimal Static Strategy

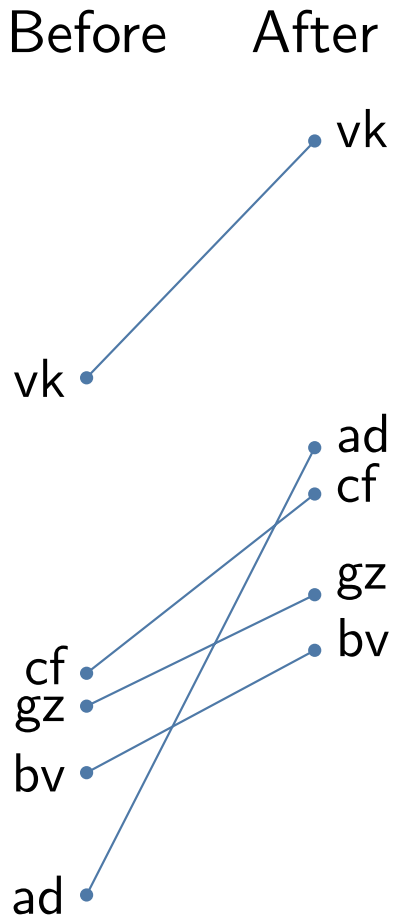
- Not possible with a lab experiment
- Beta family (any mode and concentration)
- Objective: minimize the gap with VK



Optimal Static Strategy

- Not possible with a lab experiment
- Beta family (any mode and concentration)
- Objective: minimize the gap with VK





https://econgames.org

Complete stack

- VPS at `transip.nl`, €10/month
- Arch Linux
- PostgreSQL, Apache, PHP, TeX
- Mediawiki
- Custom maillist

Future plans

- Maintenance
- Templates (lit., affil.)
- Python framework integration

```
#+FILETAGS: :@home:
* EconGames Wiki
** DONE Domain Name
   - competition-games.org
   - econgames.org
** DONE MediaWiki
   - [X] "In development" page
   - [X] PostgreSQL
   - [X] PHP with required extensions
   - [X] Configure Apache, test PHP
   - [X] Add SSL (otherwise private section does not make sense!)
     - [X] TransIP certificate for econgames.org (need to read on on security prefs)
     - [X] HSTS
     - [X] Restrict SSL Suite
     - [X] Something is not right with session tokens/id (restart server every night?)
     - [X] ipv6 is not working, strangely
   - [X] MediaWiki setup
   - [X] Private and public spaces
   - [X] Contact the colleagues
** CANC Math / Mathoid...
** DONE Math / Texvc
   - [X] install texvc
   - [X] test formulas rendering
   - [X] debug the hell out of it
   - [X] email with announcement
** DONE Email Notifications
   - [X] install an mta (postfix)
   - [X] configure rdna (ipv4 only at the moment)
   - [X] bind postfix smpt to econgames.org ip address
   - [X] test mta (how?)
   - [X] enable email functionality
   - [X] news page (so that others can watch it)
   - [X] inform everyone
** TODO Security
   - [ ] chroot php-fpm
   - [ ] send mail with ssl
   - Always see observatory.mozilla.org
** TODO Various
   - [ ] Update email to use ipv6
   - [X] name rewriting scheme
   - [X] website seems to be inaccessible via IPv6
** DONE Search engines
   - [X] register with Bing
   - [X] register with Google
   - [X] add econgames to Google sites
** DONE Backup
** DONE Mailing list
   - [X] play with postfix to discover how stuff works
   - [X] simple mailing list with php (all necessary soft already
     installed, plus I need to learn php for the wiki)
   - [X] check if attachments are handled correctly
   - [X] mailing list label
   - [X] mediawiki interface
   - [X] alias list from postgresql
   - [X] access control
   - [X] save history for the future web interface
   - [X] backup
   - [X] add everyone to the general@econgames.org
   - [X] send a test request around
   - [X] check that everyone has it working correctly
   - [X] redirect postmaster@econgmaes.org to andrei.dubovik@gmail.com
   - [X] miscellaneous
     - [X] comment the code
     - [X] sql injection attack
   - [X] Option to subscribe to the default mailing list for new accounts
```

Programming Framework

Python

- Most popular interactive programming language
- NumPy, SciPy
- Allowing C/C++, R, Matlab, Mathematica makes it difficult for other participants to simulate the whole population
- Caveats
  - No enforced boundaries between functions
  - Participants' code can be slow

Future: own git server

- Single sign-on with econgames.org
- No public access till the tournament
- Opens a possibility for a real-time tournament
- Not trivial

Good Participant

```
def player1():
    return 4

def player2():
    return 5

def play():
    p1 = player1()
    p2 = player2()
    i, p = (1, p1) if p1 < p2 else (2, p2)
    print('Player {} wins €{}'.format(i, p))
```

Bad Participant

```
import inspect

def player1():
    return 4

def player2():
    f = inspect.stack(0)[1].frame
    p1 = f.f_locals['p1']
    return p1-0.01

def play():
    p1 = player1()
    p2 = player2()
    i, p = (1, p1) if p1 < p2 else (2, p2)
    print('Player {} wins €{}'.format(i, p))
```

Very Bad Participant

```
import inspect
import ctypes

def player1():
    return 4

def player2():
    f = inspect.stack(0)[1].frame
    f.f_locals['p1'] = 1000001
    ctypes.pythonapi.PyFrame_LocalsToFast(
        ctypes.py_object(f), ctypes.c_int(0))
    return 1000000

def play():
    p1 = player1()
    p2 = player2()
    i, p = (1, p1) if p1 < p2 else (2, p2)
    print('Player {} wins €{}'.format(i, p))
```

Motivation

- Fun to organize and play
- A source for new research ideas
- Subjects
  - Capacity constraints
  - Search costs
- Practical applications
  - Antitrust assessments
  - IO education
- Theory development
  - Validating the Nash equilibrium concept
  - Discovering evolutionary stable strategies
- Massively multiplayer online real-time strategy games

Tournament 2

- Summer 2019 (preliminary)
- The same Bertrand-Edgeworth game
- 3 prizes: long prize, short prize, and spot prize
- Preparation
  - Write a short report about the Erasmus tournament
  - Student organizations, considering IO journals
  - Update Python code (separate game and simulation, etc.)
- Everybody is welcome
  - Send me an email to register
  - Participate in the tournament
  - Participate in the organization