Efficient Algorithms and Their Applications

Monika Henzinger

universität
wien

Efficient algorithms

- **Topics**: Algorithm analysis and complexity, data structures, combinatorial optimization, algorithmic game theory

- **Main conferences**: STOC, FOCS, **SODA**, ICALP, **ESA,** EC
  - SODA 2012: 523 submissions, 138 acceptances

⇒Thriving theory community

Impact?

- **Personal experience at Google:**
  - Undergraduate algorithms knowledge is bread-and-butter
    - Data structures in C++ STL
  - Current research topics generate little interest
    - Except for algorithmic game theory
    - Very different from Information Retrieval

- **Possible reasons:**
  - "Life" does not require complicated algorithms often
  - The research community is working on the wrong problems
  - The research community is using the wrong evaluation criteria

Evaluation criteria

- **Knuth:** ". . .we want *good* algorithms in some loosely defined aesthetic sense. One criterion . . . is the length of time taken to perform the algorithm . . .. Other criteria are adaptability of the algorithm to computers, its simplicity and elegance, etc"

- I would add: "and how it works on real-life data sets."

Task scheduling in Google cloud

- Given cluster of 1000s of machines with different properties
- **Online setting**:
  - Whenever a new job arrives it is partitioned into 100s-1000s of task that have to be assigned to machines
  - To which machines to assign the tasks?
- **Theory research**: Multi-processor scheduling
  - Offline problem is NP-hard
  - Large set of approximation algorithms: First-fit, best-fit, sum-of-squares, worst-fit
  - Different evaluation criteria: Minimize make-span, minimize waiting time, …

Practical constraints

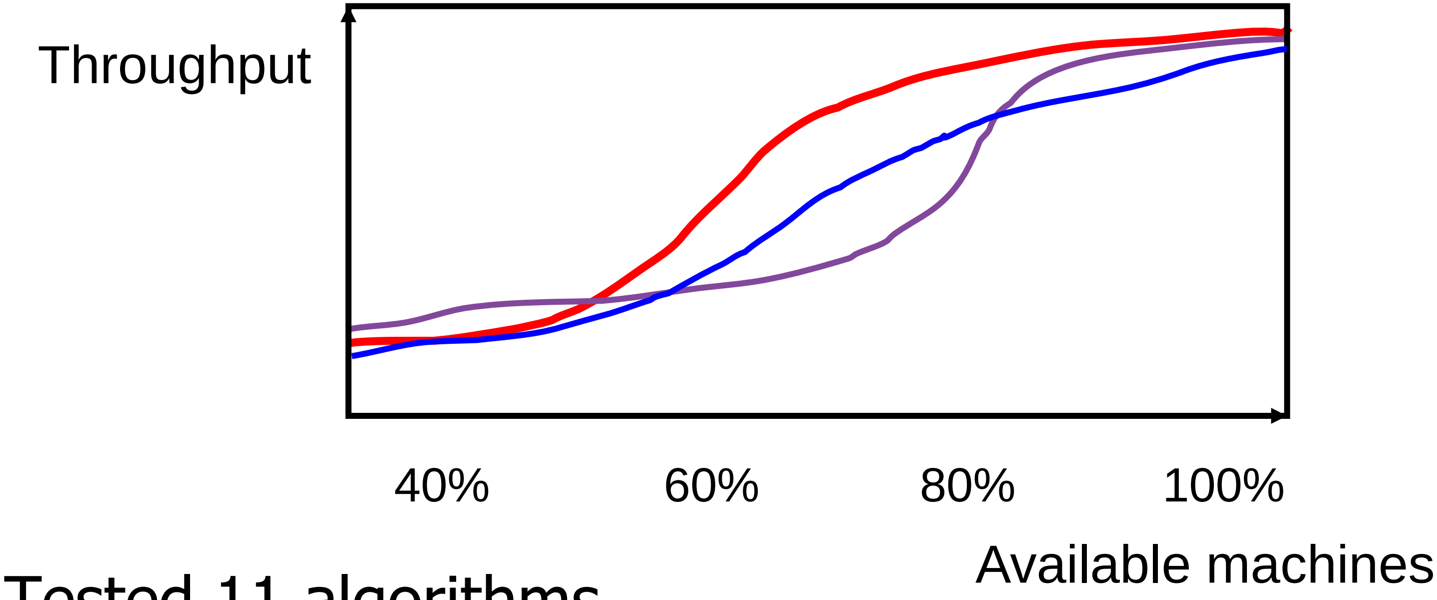
- **Evaluation criterion 1:**
  - Maximize resource utilization (throughput)
- Collected data from 8 data centers:
  - Distribution of tasks lengths and arrival rates and times very inconsistent
  - *None were close to capacity, all algorithms performed the same*
- **Evaluation criterion 2:**
  - Ability to handle more jobs on same cluster of machines in same time

⇒ None of the previous evaluation measure fits

⇒ Simulation hard since increase in load cannot be easily simulated

Practical solution

- Randomly select 40%, 60%, 80% … of the machines and check in simulator how many tasks complete in fixed time interval



- Tested 11 algorithms
- 3 evaluation criteria: throughput, time spent for useful work, machine fragmentation

Task scheduling in Google cloud

- Thorough study
- Best-Fit clearly best

⇒ Has been running in Google cloud since 2009

⇒ Google presented this work at 2011 Faculty Summit

… but we never published it

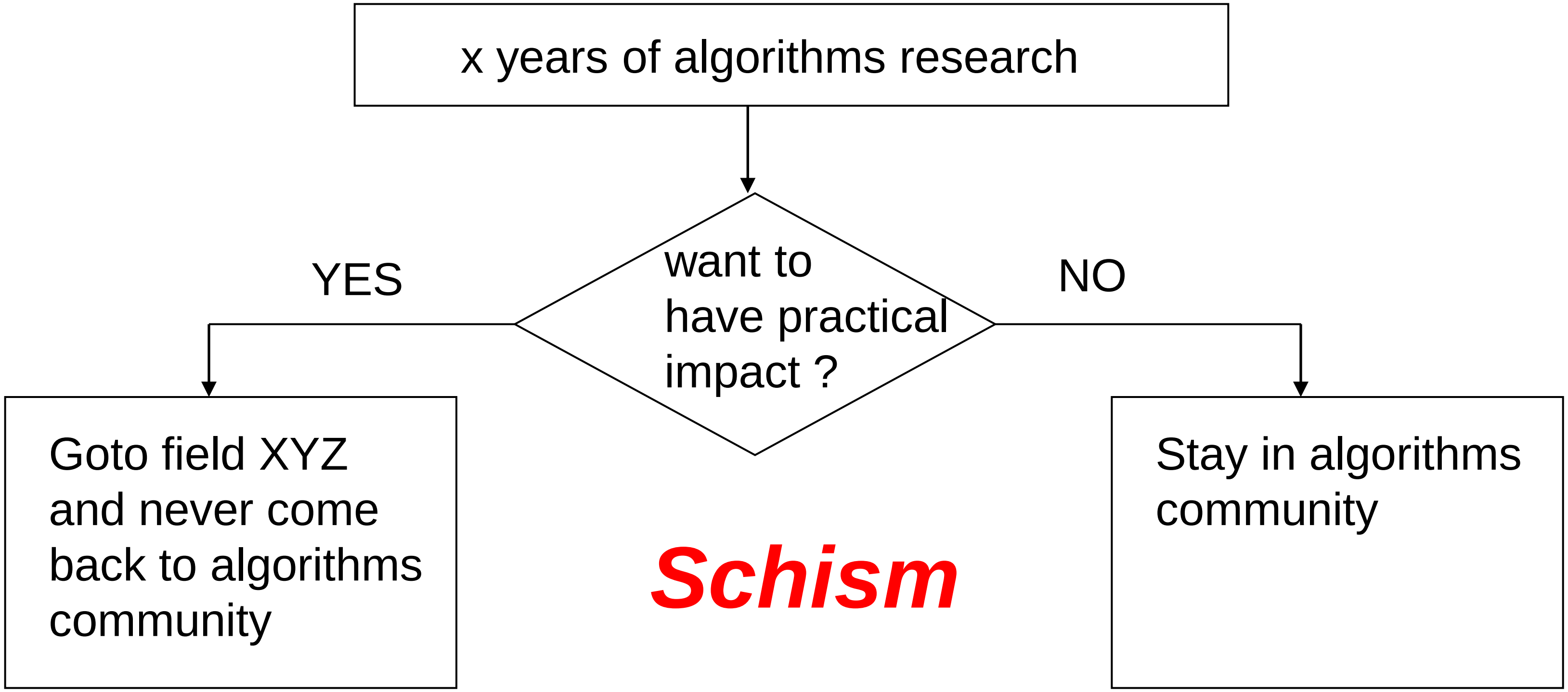
Applied algorithms work

- Publication venues:
  - Workshop of Experimental Algorithms
    - Not highly regarded
  - SODA
    - Paper needs to mostly contain new theory and then experimental results confirming them
  - Conferences in the field, e.g. Cloud Computing Conference
    - Needs to know the conference

⇒ No good option available!

⇒ Disincentive to do applied algorithms work

Life of algorithms researcher



x years of algorithms research

want to have practical impact ?

YES

NO

Goto field XYZ and never come back to algorithms community

Stay in algorithms community

Schism

⇒ Big drain for the algorithms community

⇒ Feedback from the field is missing

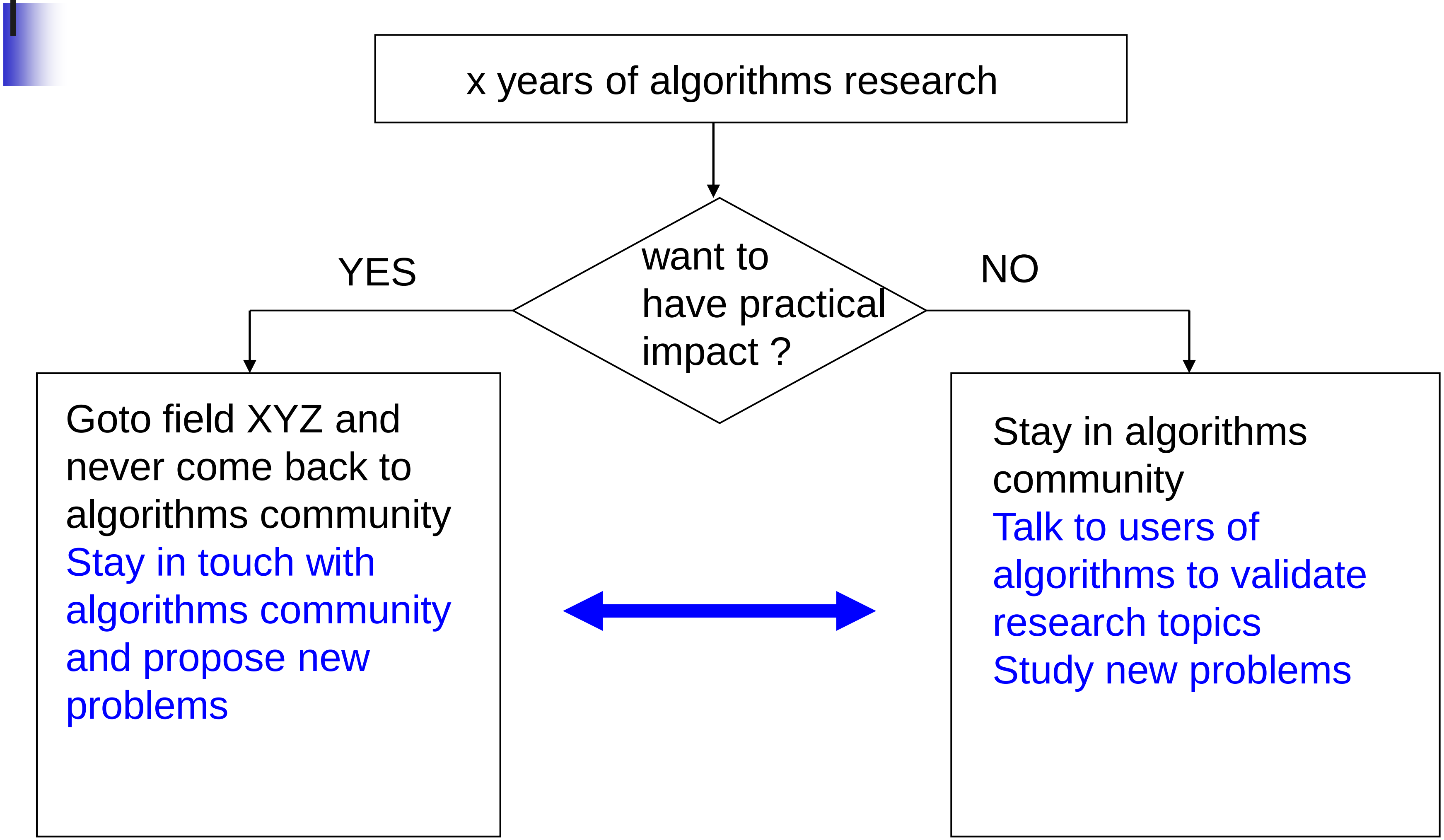
Other communities

- Example: Information Retrieval community
  - Theoretical as well as practical work is valued
  - Standard data sets for experiments
  - Continuously looking for input from computer scientists in the field and adapting set of problems of interest

⇒ Algorithms community could learn from them

⇒ Every algorithms researcher can learn from them

Life of algorithms researcher



x years of algorithms research

want to have practical impact ?

YES

NO

Goto field XYZ and never come back to algorithms community
Stay in touch with algorithms community and propose new problems

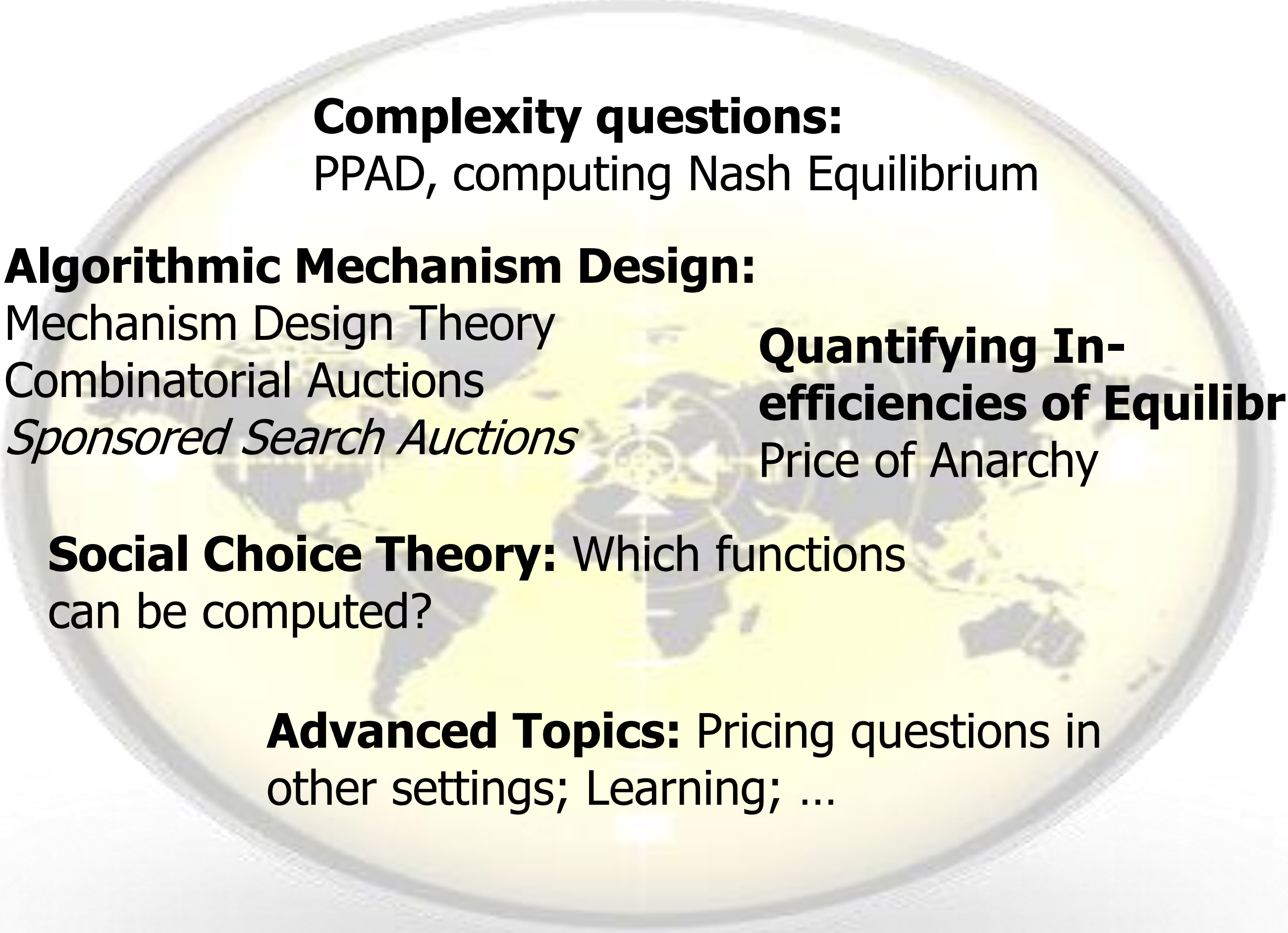
Stay in algorithms community
Talk to users of algorithms to validate research topics
Study new problems

Success Story: Algorithmic Game Theory

Design algorithms in strategic environments:

- Decisions need to made with self-interested participants
  - *Example: m items need to be assigned to n bidders*
- Goal: Conditions to be fulfilled by the outcome, e.g. social-optimum solution
  - *Example: Items are given to bidders that value them most*
- Equilibrium concept:
  - Truthfulness: The participants achieve the highest utility if they give truthfully input (independent of the input of the others)
  - Nash equilibrium: At the given solution no participant achieves higher utility by giving a different input (dependent on the input of the others)
  - …

Algorithmic Game Theory: Big Picture



Complexity questions:
PPAD, computing Nash Equilibrium

Algorithmic Mechanism Design:
Mechanism Design Theory
Combinatorial Auctions
Sponsored Search Auctions

Quantifying Inefficiencies of Equilibria:
Price of Anarchy

Social Choice Theory: Which functions can be computed?

Advanced Topics: Pricing questions in other settings; Learning; …

Algorithmic Game Theory

- Close interaction with economists and companies such as web search engines that run internet auctions
- Quickly adapting to new problems of interest
- Running time is secondary, but equilibrium concepts “enforce” simple algorithms

Schism in teaching

- Typical 1st year bachelor algorithms class
  - Class: Definitions, algorithms and proofs
  - Homework: Proofs
  - Exams: More proofs
  - Applications briefly mentioned

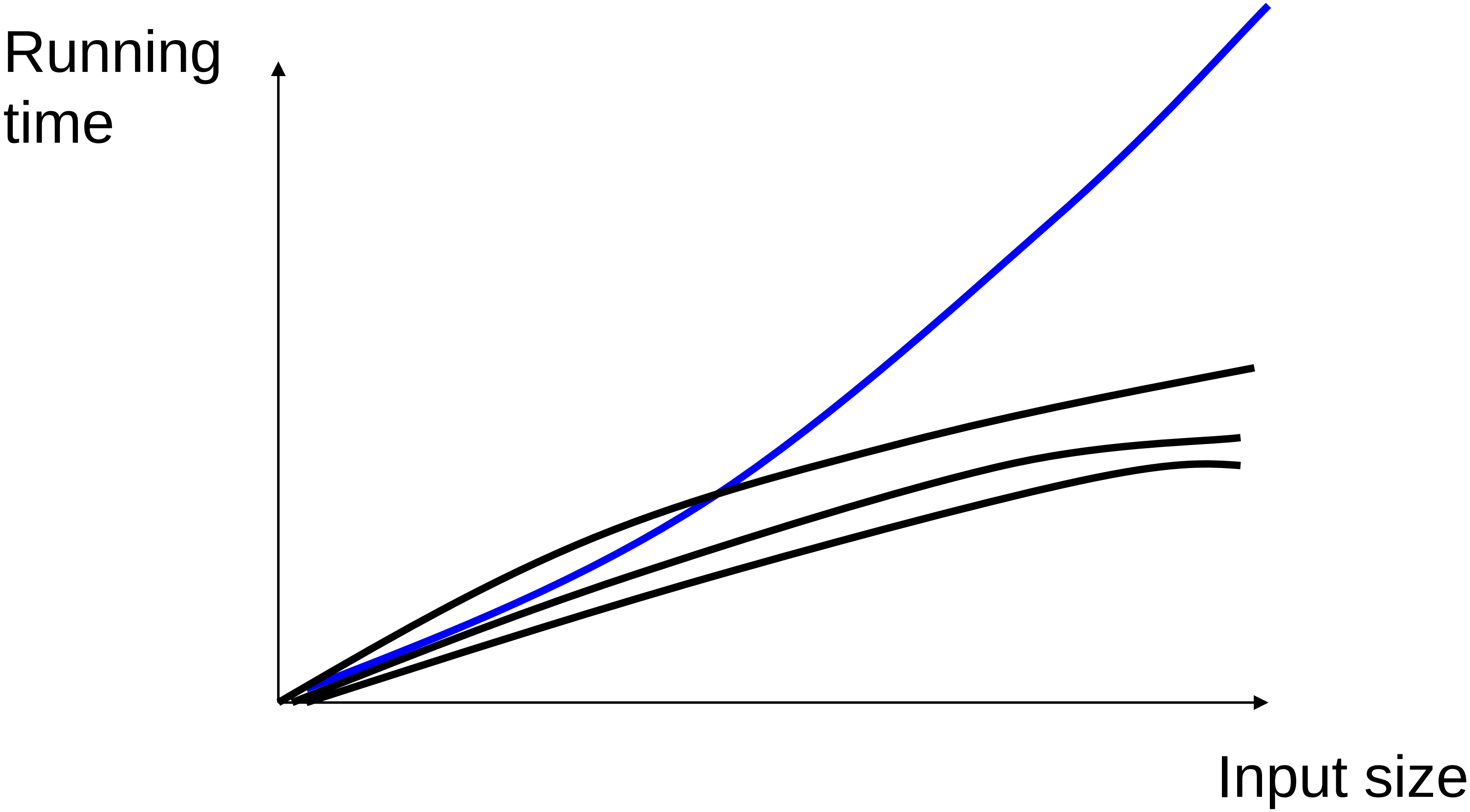
$\Rightarrow$ Theory-only class

Schism in teaching (cont.)

- 1st year bachelor algorithms class at University of Vienna
  - Designed by database/hpc researchers
  - Class: Definitions and algorithms
  - Homework: Semester-long implementation project
    - Evaluated in comparison to other students
  - Exam: Algorithms

⇒ Proof-free class

Feedback to the student



Algorithms Teaching

- Ideal 1st year bachelor algorithms class
  - Class: Definitions, algorithms, and proofs
  - Homework: Proofs and implementation
    - Designed with applied researcher
  - Exams (with low weight): Algorithms and proofs

⇒ Overcome the schism!

Algorithms Teaching

- Important material missing in basic algorithms class:
  - Data streams
  - Online algorithms
  - Randomized algorithms
  - Approximation algorithms

⇒ Should be taught in second required algorithms class