Pavel Chuprikov

pavel.chuprikov@usi.ch
https://www.inf.usi.ch/postdoc/chuprikov · 0000-0002-6673-1143 · Updated on August 2, 2022

Work Experience

Università della svizzera italiana — **Lugano, Switzerland**
Postdoc — *January 2020–present*
Computer Systems Institute

IMDEA Networks Institute — **Madrid, Spain**
Research assistant — *February 2016–January 2020*
Networked Systems Group
Research title: Network algorithms
Supervisor: Kirill Kogan — *IMDEA Networks Institute*

JetBrains — **St. Petersburg, Russia**
Software developer — *September 2015–February 2016*
HoTT and Dependent Types Group

JetBrains — **St. Petersburg, Russia**
Software engineering intern — *June–August 2015*
HoTT and Dependent Types Group

JetBrains — **St. Petersburg, Russia**
Software engineering intern — *July–September 2014*
Group for Verification of Operational Transformation

Transas New Technologies — **St. Petersburg, Russia**
Software Engineering Intern — *November 2011–September 2013*
3D visualization department

Education

Steklov Institute of Mathematics at St. Petersburg — **St. Petersburg, Russia**
PhD student, Laboratory of Mathematical Logic — *October 2015–October 2019*
Thesis: Theoretical and empirical analysis of fundamental bottlenecks in networking and distributed computing
Supervisor: Sergey Nikolenko — *HSE University, Steklov Institute of Mathematics at St. Petersburg*
Supervisor: Kirill Kogan — *IMDEA Networks Institute*
Defended on: 21st of November, 2019 — *HSE University*

ITMO University — **St. Petersburg, Russia**
Student — *July 2017*
School on practice and theory of concurrent computing, summer school

St. Petersburg Academic University — **St. Petersburg, Russia**
Master of Science student, Applied Mathematics and Physics, MIT Dept. — *October 2013–June 2015*
Thesis: Priority queueing with multiple packet characteristics — *16th of June, 2015*
Supervisor: Sergey Nikolenko — *HSE University, Steklov Institute of Mathematics at St. Petersburg*

Bioinformatics Institute — **St. Petersburg, Russia**
Student, Algorithmic Bioinformatics — *September 2014–May 2015*

ITMO University — **St. Petersburg, Russia**
Bachelor of Science student, Applied Mathematics and Computer Science, CT Dept. — *September 2009–June 2013*
Thesis: Realistic rendering of shoreline waves in real-time — *20th of June, 2013*
Supervisor: Alexey Romanov — *Transas New Technologies*

Publications

Journals:

- **P. Chuprikov**, A. Davydow, K. Kogan, A. Sirotkin, S. I. Nikolenko. *Formalization and taxonomy of compute-aggregate problems for cloud computing applications*, Computer Networks, 2021
- A. Davydow, **P. Chuprikov**, S. I. Nikolenko, K.Kogan. *Competitive buffer management for packets with latency constraints*, Computer Networks, 2021
- V. Demianiuk, S. I. Nikolenko, **P. Chuprikov,** K. Kogan. *New Alternatives to Optimize Policy Classifiers*, IEEE/ACM Transac-

tions on Networking, 2020
- **P. Chuprikov**, S. I. Nikolenko, K. Kogan. *Towards declarative self-adapting buffer management*, ACM SIGCOMM CCR, 2020
- **P. Chuprikov**, S. I. Nikolenko, A. Davydow, K. Kogan. *Priority Queueing for Packets with Two Characteristics*, IEEE/ACM Transactions on Networking, 2018

Conference proceedings:
- P. Jahnke, V. Riesop, P.-L. Roman, **P. Chuprikov**, P. Eugster. *Live in the Express Lane*, USENIX ATC 2021
- **P. Chuprikov**, V. Demianiuk, S. Gorinsky. *PREDICAT: Efficient Packet Classification via Prefix Disjointness*, IEEE ICCCN 2021
- K. Kogan, A. Davydow, S. Nikolenko, **P. Chuprikov**, V. Demianiuk. *SRPT-based congestion control for flows with unknown sizes*, IFIP Networking 2021
- **P. Chuprikov**, K. Kogan. *How to network delay-sensitive applications*, IFIP Networking 2021
- V. Demianiuk, S. Nikolenko, **P. Chuprikov**, K. Kogan. *New Alternatives to Optimize Policy Classifiers,* ICNP 2018
- **P. Chuprikov**, A. Davydow, K. Kogan, S. Nikolenko, A. Sirotkin. *Formalizing Compute-Aggregate Problems in Cloud Computing,* SIROCCO 2018,
- **P. Chuprikov**, K. Kogan, S. I. Nikolenko. *How to implement complex policies on existing network infrastructure,* SOSR 2018,
- K. Kogan, S. I. Nikolenko, V. Demianiuk, **P. Chuprikov**, A. Davydow. *Personal insights on three research directions in networked systems*, COMSNETS 2018
- **P. Chuprikov**, K. Kogan, S. I. Nikolenko. *General ternary bit strings on commodity longest-prefix-match infrastructures,* ICNP 2017
- A. Davydow, **P. Chuprikov**, S. Nikolenko, K. Kogan. *Throughput Optimization with Latency Constraints*, INFOCOM 2017
- S. Sinchuk, **P. Chuprikov**, K. Solomatov. *Verified operational transformation for trees*, ITP 2016
- **P. Chuprikov**, S. Nikolenko, K. Kogan. *On Demand Elastic Capacity Planning for Service Auto-scaling*, INFOCOM 2016
- **P. Chuprikov**, S. Nikolenko, K. Kogan. *Priority Queuing with Multiple Packet Characteristics*, INFOCOM 2015

Posters:
- **P. Chuprikov**, A. Davydow, K. Kogan, S. I. Nikolenko, A. V. Sirotkin. *Planning in compute-aggregate problems as optimization problems on graphs*, ICNP 2017

Approved research projects

- *Towards comprehensive network monitoring and management*, 50'000 CHF, Hasler foundation, April 2020–March 2021

Supervision

- *Fault tolerance and performance of atomic commit algorithms* (co-supervision with P. Eugster), Bachelor thesis at USI Lugano
- *Confidentiality-preserving processing of symmetrically-encrypted streaming data*, Master thesis at ITMO University

Teaching and other scientific activities

St. Petersburg Academic University — **St. Petersburg, Russia**
Teaching assistant, Combinatorics and Graph Theory — *September 2014–January 2015*

- Ad-hoc reviewer for ACM/IEEE Transactions on Networking, IEEE INFOCOM, IEEE ICNP, ESOP, ACM PODC

Prizes, awards, fellowships

- Yandex scholarship to support research, September 2013–June 2015.
- Third degree award at ACM ICPC NEERC 2011.
- Third place at International High Performance Programming Contest 2013.
- Third degree award at Putnam competition at St. Petersburg, 2013.

Research interests

- Distributed systems
- Formal methods and programming languages
- Packet classification
- Network congestion control
- Online algorithms and competitive analysis
- Admission control and buffer management

Personal skills

Computing: Linux, C++, Python, Rust, Git, LaTeX (TikZ), Coq, Nix, Haskell, Agda
Computer Networks: P4, OMNET++ (INET), NS2, OpenFlow, Mininet
Languages: Russian (Native), English (Fluent), Italian (Elementary), Spanish (Beginner), German (Beginner)
Other: juggling, cycling, electrical engineering hobbyist

Major scientific achievements

Efficient representation of packet classifiers on longest-prefix match infrastructure

Packet classification allows the network to distinguish among different traffic types and process each packet accordingly. Due to the growing diversity of network services, complexity of packet classification is also increasing, and it is now necessary to consider predicates over many packet header fields when determining traffic type. Traditionally, the network only performed packet forwarding, where the decisions was based on destination address only. The main hardware solution to multi-field classification, *ternary content-addressable memory* (*TCAM*), is expensive and power hungry.

Pavel's work "General ternary bit strings on commodity longest-prefix-match infrastructures" (co-authored with K. Kogan and S. Nikolenko) exploits flexible key-construction mechanism available at network devices to adapt TCAM-targeted packet classifiers to commodity mechanisms implementing *longest-prefix-match* (*LPM*) classification. The work presents an efficient criterion for checking whether a packet classifier can be represented in LPM form. If the criterion fails for a given classifier, the classifier can be partitioned into several groups, each in LPM form. Both minimizing the number of groups and maximizing the number of LPM-represented entries are solved optimally by a discovered connection to order theory.

Self-adapting buffer management

The congestion control at end hosts has been shown to benefit strongly from in-network support. As a result, network appliances now ubiquitously deploy a set of policies for congestion-aware buffer management. Unfortunately, policies behave differently when presented with different types of network traffic and different performance objectives. The parameter tuning is often necessary for optimal result, and if done wrong, use of such policies can even worsen network performance.

In work "Towards declarative self-adapting buffer management" (co-authored with S. Nikolenko and K. Kogan), Pavel has presented a system that does policy selection locally and in a *fully automated fashion*. The only input from the network administrator is a *reward function*, which should be able to evaluate the behavior of the policy over a given interval of time. Key ingredient of the approach is the observation that the problem is almost exactly a well-known multi-armed bandit problem from machine learning literature. A hierarchy of traffic assumptions and corresponding adaptation algorithms are then investigated in the work. The algorithms are compared using *RED* and *CoDel* policies on synthetic benchmarks.

Buffer management policies with value and processing requirements

Buffer management decides which packets to admit, process, and transmit, influencing the overall performance of the network element. The increasing number of services running at network elements demands flexible objectives to accommodate traffic diversity. As a result, every network packet gets associated with an intrinsic "value", i.e., relative importance of transmitting that packet. Also, the processing requirements of individual packets become highly-non uniform. Both characteristics need to be accounted for in buffer management under the uncertainty of future packet arrivals.

Pavel has applied worst-case competitive analysis to the problem of buffer management with two characteristics in his work "Priority Queueing for Packets with Two Characteristics" (co-authored with S. Nikolenko, A. Davydow, and K. Kogan). It has been shown that ultimately online algorithms are *not* competitive in the general setting, meaning they can behave arbitrarily worse that the optimal *clairvoyant* algorithm. On the positive side, in the special case of only two possible values (e.g., representing "commodity" and "gold" traffic), an algorithm preferring higher value, breaking ties by lower processing is within factor $(1 + \frac{W+2}{V})$ of optimal, where W is maximal processing and V—maximal value (normally, $V > W$).

Network service models for delay-sensitive applications

The rise of interactive analytical services led to a growing number of applications requiring the data to arrive before a given deadline or not at all. The issue is that the existing service models and most works on low latency communication put reliability in front of delay guarantees wasting network resource on transmitting outdated and useless packets

In his work titled "How to Network Delay-Sensitive Applications" (co-authored with K. Kogan) Pavel draws inspiration from the bounded-delay setting widely used in buffer management literature to reconsider existing service models. The defining feature of that model is precisely that late packets add nothing to the optimization objective. It has been shown that *infinite buffer sizes* are sufficient for analyzing algorithm performance under such a service model as there shown to exist a universal performance-preserving transformation to *finite* buffers. A different universal transformation has been also introduced to transfer result from throughput-based objective to a goodput-based objective. Moreover, the effect on the latter objective has been formally quantified for different degrees of coupling between buffer management and congestion control.

Compute-aggregate planning

Compute-aggregate computations are now ubiquitous in distributed data analytics due to their scalability. Unfortunately, the execution of this computations loads the network heavily causing congestion and TCP-incast problems.

Pavel's solution, presented in "Formalization and taxonomy of compute-aggregate problems for cloud computing applications" (co-authored with A. Davydow, K. Kogan, et al.) takes advantage of intermediate aggregation and splits the problem in two phases: planning and execution. The planning phase relies on a novel piece of information from application, namely *aggregation size function*. A taxonomy of aggregation size functions is proposed for selecting appropriate planning strategies.