Network Algorithms Ex Machina
Towards Self-Driving Networks

Andreas Blenk*, Patrick Kalmbach*, Johannes Zerwas*, Wolfgang Kellerer*, Stefan Schmid†

Technical University of Munich*, University of Vienna†

Self-Driving Softwarized and Virtualized Networks

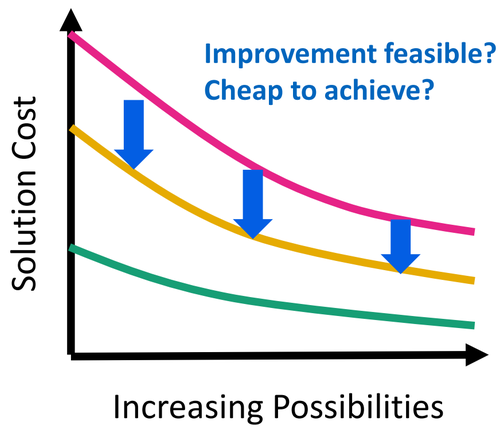
Context: Networks becoming more and more complex

Problem: Human struggle in designing new solutions: network protocols, topology design, or algorithms

Goal:

- Make networks self-X (self-optimized, self-adaptive, self-monitoring, ...)
- Enabling technologies: Software-Defined Networking (SDN) and Network Virtualization (NV)

This poster's challenge: How to exploit given flexibilities introduced by SDN and NV while facing hard network algorithm problems



Optimal Algorithm:
- Improves solution quality given more flexibilities
- However, expensive, exponential runtime

Heuristic Algorithm:
- Can exploit flexibility
- But cannot achieve optimal solution

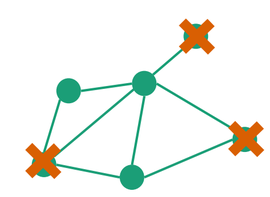
Machine Learning/Neural Computation:
- Improves solution quality
- Impact of learning time? Computational overhead? RESEARCH!

Contribution presented in this poster:

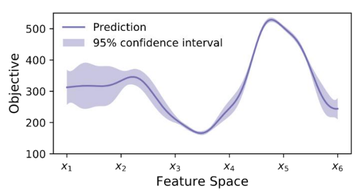
- **o'zapft is:** Supervised learning using algorithm data [1,3]
- **NeuroViNE:** Using Hopfield neural network for efficient pre-processing [2,3]

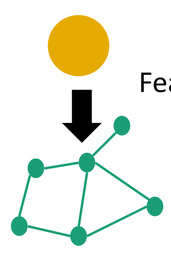
Speeding Up Algorithms

Exemplary Problem: Find suitable location for virtualized network cache



Excluded nodes





(1) Shrink Search Space | **(2) Estimate Cost** | **(3) Predict Feasibility**

Future Vision

Complete Data-Driven Algorithm Design

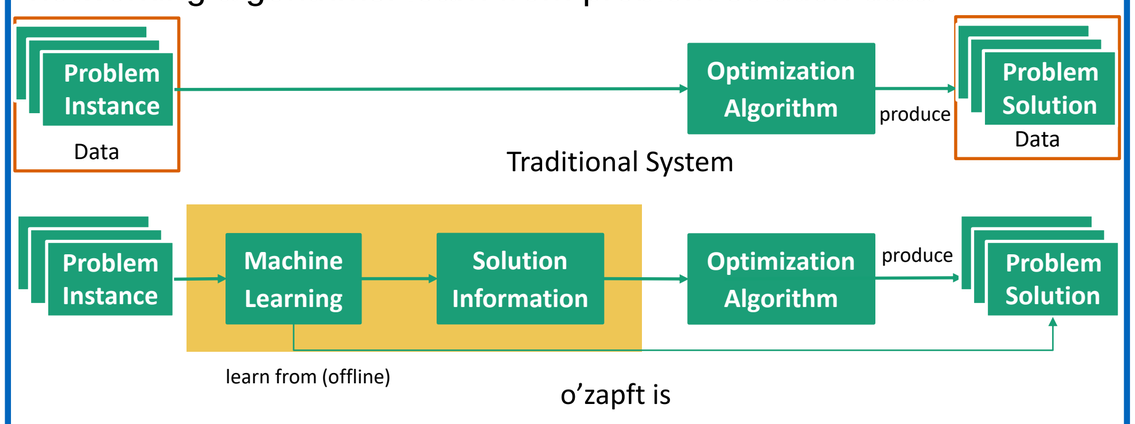
Problem → Machine → Algorithm

Related work has demonstrated automated design for Congestion Control (TCP)

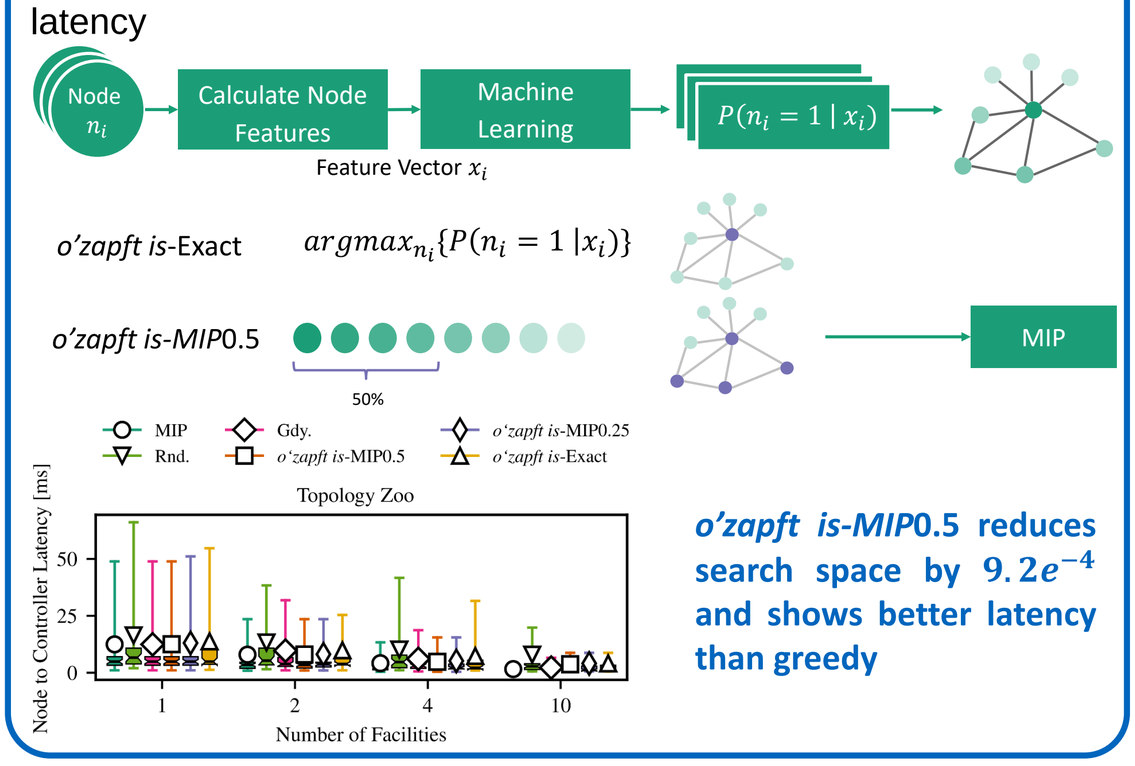
- K. Winstein and H. Balakrishnan, "Tcp ex machina: Computer-generated congestion control," in Proc. ACM SIGCOMM, 2013
- F. Y. Yan et al., "Pantheon: the training ground for Internet congestion-control research," in Proc. USENIX NSDI, 2018

Supervised Learning

o`zapft is: A supervised learning approach to speed up networking algorithms: learn from problem/solution data



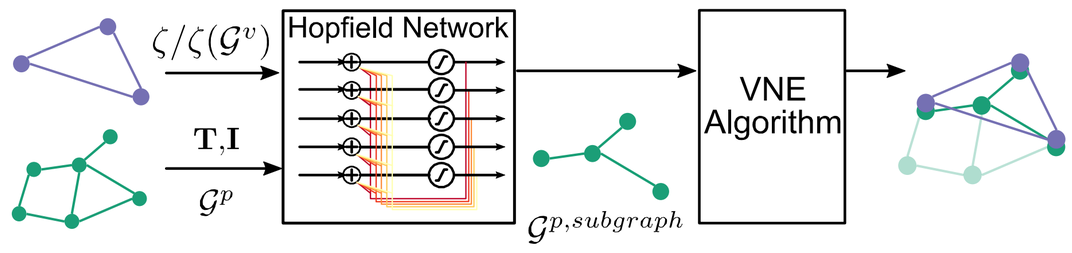
Problem: Place facility on network node; minimize maximum latency



***o'zapft is-MIP*0.5 reduces search space by $9.2e^{-4}$ and shows better latency than greedy**

Neural Computation

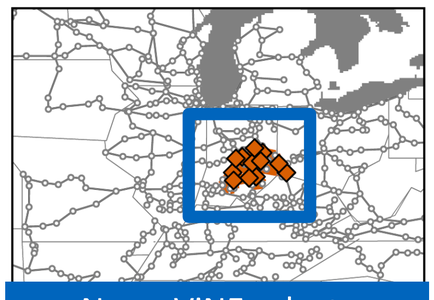
NeuroViNE: A neural pre-processor using energy-based Hopfield neural networks for your **Virtual Network Embedding** algorithm



Neural computation | Reuse existing VNE algorithms

Parallel computation | Implementable on hardware

Selected nodes by VNE heuristic lead to long paths!



NeuroViNE selects substrate nodes close to each other!

Hopfield network solution provides nodes with high capacity close to each other

References

[1] A. Blenk, P. Kalmbach, S. Schmid, and W. Kellerer, "o'zapft is: tap your network algorithm's big data!" In Proc. ACM SIGCOMM Workshop Big-DAMA, Los Angeles, CA, USA: ACM, August 2017, 19–24.

[2] A. Blenk, P. Kalmbach, J. Zerwas, M. Jarschel, S. Schmid, and W. Kellerer, "NeuroViNE: A Neural Preprocessor for Your Virtual Network Embedding Algorithm," in Proc. IEEE INFOCOM, Best-In-Session Presentation Award, Honolulu, HI, USA, April 2018, 1–9.

[3] A. Blenk, "Towards Virtualization of Software-Defined Networks: Analysis, Modeling, and Optimization", PhD Thesis, Technische Universität München, 2018.

Contact: andreas.blenk@tum.de| Chair of Communication Networks (LKN) | Technische Universität München (TUM), Arcisstr. 21, 80333 Munich, Germany | www.lkn.ei.tum.de